Proceedings of the Final Workshop of the Grid Projects of the Italian National Operational Programme 2000-2006 Call 1575

Catania, Italy
10-12 February 2009

Edited by
Consorzio COMETA

UNIONE EUROPEA

PON Ricerca Scientifica Sviluppo Tecnologico Alta Formazione 2000 - 2006

Ministero dell' Università e della Ricerca

Property of:

Consorzio COMETA
Via Santa Sofia 64
I-95123 Catania-Italy
Year 2009

http://www.consorzio.cometa.it

http://agenda.ct.infn.it/conferenceDisplay.py?confId=87

ISBN: 978-88-95892-02-3

Disclaimer: The articles presented on this proceedings book are for international purposes only. Consorzio COMETA does not accept any responsibility, loss, injury or legal liability resulting from the use or misuse of the information presented in the articles published here. The article authors' views do not necessarily represent or endorse the views of Consorzio COMETA.

Proceedings of the

FINAL WORKSHOP OF GRID PROJECTS FUNDED BY "PON RICERCA 2000-2006, AVVISO 1575"

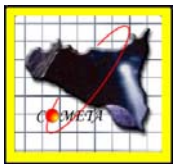
Catania, Italy, 10-12 February 2009

Editors:

Roberto Barbera

Marcello Iacono Manno

Marco Fargetta



Scientific Committee

Roberto Barbera (Univ. di Catania e Consorzio COMETA) – Chair
Giuliano Laccetti (Università "Federico II" di Napoli)
Giuseppe Longo (Università "Federico II" di Napoli)
Alberto Masoni (INFN e Consorzio COSMOLAB)
Silvio Migliori (ENEA)
Antonio Puliafito (Univ. di Messina e Consorzio COMETA)
Andrea Quintiliani (ENEA)
Salvatore Sciortino (INAF e Consorzio COMETA)
Gianluigi Zanetti (CRS4 e Consorzio COSMOLAB)

Organizing Committee

Giuseppe Andronico (INFN e Consorzio COMETA) – Chair
Simona Arriva (Consorzio COMETA)
Francesca Nicotra (Consorzio COMETA)
Silvio Pardi (Università "Federico II" di Napoli)
Carolina Rapicavoli (Consorzio COMETA)

Programme

Tuesday 10 February 2009

Inaugural Session - Aula Magna (08:30-18:30)

Conveners: Catara F., Murli A.

10:30 Welcome
11:30 Mazzuccato M., *The European Grid Initiative: present and future*
12:00 Mazzuccato M., *L'Infrastruttura Grid Italiana: stato attuale e prospettive future*
12:30 Barbera R., *GRISU': la Grid del Sud*

14:30 Valente E., *La rete GARR: stato attuale e prospettive future*
15:00 Migliori S., *Presentazione generale sui risultati scientifici del progetto*
16:00 Zanetti G., *Presentazione generale sui risultati scientifici del progetto Cybersar*
16:30 Pappalardo G., *Presentazione generale sui risultati scientifici del progetto PI2S2*
17:00 Merola L., *Presentazione generale sui risultati scientifici del progetto SCoPE*
17:30 Aloisio G., *Il progetto SPACI: storia passata e prospettive future*

Wednesday 11 February 2009

Session 1 – Physical Sciences (Micro-cosm, Macrocosm, Earth) – Aula Magna (09:30-14:30)

Convener: Merola L.

09:30 Pasian F., *Data and compute-intensive applications in astrophysics: the Virtual Observatory*
10:00 Carlino G., *Distributed analysis for the ATLAS Experiment in the S.Co.P.E Project*
10:20 Antonuccio V., *Project GALCOS: GALaxy Formation within a COSmological Volume*
10:40 Brescia M., *The DAME/VO-Neural infrastructure: an integrated data mining system support for the e-science community*
11:00 Pusceddu G., *Implementation of two limited area ensemble meteorological systems in a GRID environment*

11:50 Mangiagli S., VTA: *GRID based Thermal Images Processing for volcanic activity monitoring*
12:10 Sannino G., *Impact of a two-way grid refinement at the Strait of Gibraltar on the thermohaline circulation of the Mediterranean Sea*

12:30 Mazzarella G., *Synthesis of EBG Surfaces Using Evolutionary Optimization Algorithms*
12:50 Rosato V., *Porting the MIT-Global Circulation Model on the Cell section of the CRESCO System*
13:10 Fiore S., *Efficient and Scalable Climate Metadata Management with the GRelC DAIS*

Session 2 – Life and Matter Sciences –
Aula Magna (14:30-18:30)

Conveners: Mazzarella G., Laccetti G.

14:30 Milanesi L., *Computational challenges in bio-informatics*
15:00 Cuttone G., *Dose Distributions Verifications in Medical Physics: Results Obtained Using GEANT4 and the Grid Technology*
15:20 Pappalardo F., *Grid-based cancer growth simulations*
15:40 Robertazzzi A., *Simulations of Copper Phenanthroline DNA Cleavers*

16:30 Boccia V., *The MedIGrid PSE in an LCG/gLite environment*
16:50 Aprea G.A.P., *Parallel implementations of Genetic Algorithms for Parameters Estimate of Metabolic Pathways*
17:10 Di Grazia F., *Brain Functional Synchronization on Grid*
17:30 Corso P.P., *Study of the effects of nuclear motion on High Harmonic Generation in Simple Molecules*
17:50 Celino M., *Materials for hydrogen storage: ab-initio Molecular Dynamics study in the ENEA-GRID Environment*

Thursday 12 February 2009

Session 3 – Other Sciences (Social Science, Humanities, Mathematics) –
Aula Magna (09:30-15:00)

Conveners: Migliori S.

09:30 Gargiulo C., CFD *in ambito automotive: hardware, software e capitale umano*
10:00 Giacomazzi E., *Numerical Simulation of Turbulent Combustion on CRESCO Platform*
10:20 Manca A., *Multi-Airport Dynamic Network Flow Model with Capacity Uncertainty*
10:40 La Rocca G., *ASTRA Project Achievements*
11:00 Ingrà E., *Grid2Win: The Grid for Microsoft Windows*

11:50 Santoro A., *ENEA-GRID and glite Interoperability: Robustness of SPAGO Approach*
12:10 Anedda P., *VIDA: A new Virtual Images Deployment Architecture*
12:30 Rocchi A., *New ENEA-GRID Integrated Support Services for CRESCO*
12:50 Fargetta M., *HPC Applications on the Sicilian Grid Infrastructure*

TABLE OF CONTENTS

Committees……... iii
Program……..... iv
Table of Contents ……..……... vii

CALL 1575 PROJECTS
Barbera R. et al., *The PI2S2 Project and the COMETA Consortium*……... 1
Spizuoco et al., *MAGO (Monitoring All Grid Objects)*…… 13
Merola L., *SCOPE: The S.Co.P.E. Project*……... 18

BUSINESS
Ragusa C. et al., *Experimenting With Business Grids*……..... 36
Fenu G. et al., *A Model for XaaS Financial System*……..... 44

COMPUTER FLUID DYNAMICS
Ambrosino F. et al., *OpenFOAM and Fluent Features in CFD Simulations on CRESCO High Power Computing System*……..... 52
Giacomazzi E. et al., *Numerical Simulation of Turbulent Combustion on CRESCO Platform*……..... 57
Caracciuolo L. et al., *Computational Simulations of 3D Fluid-Dynamic Processes in High Performance Computing Environment* ……..... 67
Schembri F. et al., *A Grid Computational Approach to Two-Phase Flow in Microfluidics*..... 75
Gargano F. et al., *Unsteady Separation and Navier Stokes Solutions at High Reynolds Numbers*…….....…….... 81

COMPUTER SCIENCE
Minutoli G. et al., *Enabling Applications on gLite Based Grid Using Virtual Machines*……..... 89
Chinnici M. et al, *CUDA Based Implementation of Parallelized Pollard's Rho Algorithm for ECDLP*……..... 97
Scollo G., *Results and programming techniques from the CR3x+1 project*……..... 102
Marchei E. et al, *A New Metric to Describe the Efficiency of Data Communication Networks*……..... 111
Guarnieri G. et al., *Parallel Programming in Matlab Environment on CRESCO Cluster, Interactive and Batch mode*……..... 116
Blandino S. et al., *Performance Evaluation of Parallel Algorithms on Computational Grid Environment*……..... 120
Andronico G. et al., *VisualGrid: On-Line Video Streaming for Application Control and Demonstration*..... 128
Carcangiu S. et al., *Grid Computing for Internet Traffic Engineering based on OSPF/IS-IS Protocols*……..... 132
Ingrà E. et al., *Grid2Win: the Grid for Microsoft Windows*……..... 139
Barbera R. et al., *The GENIUS Grid Portal and Robot Certificates*…..... 145
Bracco G. et al., *CRESCO HPC System Integrated into ENEA-GRID Environment*... 151

Santoro A. et al., *ENEA-GRID and gLite Interoperability: Robustness of SPAGO Approach*…..... 156
Cecchi M. et al., *The gLite Workload Management System*…… 161
Rocchi A. et al., *New ENEA-GRID Integrated Support Services for CRESCO*…….. 169
Ruzzante S. et al., *Interdependencies Among Critical Infrastructures*..... 174
Ceccanti A. et al., *VOMS: INFN Contribution to Grid Security*…..... 177
Di Stefano A. et al., *Negotiation-Based Advanced Reservation to Manage SLAs in gLite*……..... 183
Di Stefano A. et al., *A QoS-Aware Distributed Discovery Algorithm for P2P-Grid Systems*……..... 191
Bandieramonte M. et al., *A Meta-Heuristic Based Scheduling Algorithm for Grid Environments*……..... 196
Pistagna C. et al., *Dissecting BitTorrent using Grid*……..……..... 203
Anedda P. et al., *VIDA: A new Virtual Images Deployment Architecture*……..... 211
Funel A. et al., *Software Tools for Monitoring ENEA-GRID with CRESCO Cluster Integration*……..... 217
Raia S. et al., *Web Crawling on ENEA-GRID Infrastructure to perform Web Archive*……..... 223
Laccetti G. et al., *Multilevel Algorithms for Multidimensional Integration in High Performance Computing Environment*……..……..... 227
Agus M. et al., *Advances in Massive Model Visualization in the CYBERSAR Project*……..... 233
Vuerli C. et al., *Grid@Trieste: A Metropolitan Area Grid* ……..... 238
Castiglia G. et al., *Integration of Matlab® with the COMETA Infrastructure: Case Study with a Scientific Application* ……..... 245
Gaggero M. et al., HaDeS: *a Scalable Service Oriented Deployment System for Large Scale Installations* ……..... 251
Mereu G.M. et al., *EyeGrid, a webOS for Access to Distributed Resources* …….. 258
Gregoretti F. et al., *Improving Node Inter-cluster Point-to-point Communication by using the UDP Protocol and NAT Service*……..... 262
Di Stefano A. et al., *An Infrastructure supporting QoS Parameters for Composite Services Running on a Grid Environment*……..……..... 267
Giunta R. et al., *Improving the Performances of a Grid Infrastructure by means of Replica Selection Policies*……..... 273
Giunta r. et al., *Measuring Performances of Globus Toolkit Middleware Services*……..... 283

CULTURAL HERITAGE

Avanzo G. et al., *The ASTRA Project Achievements: The Reconstructed Greek Epigonion*……..…… 290
D'Abate et al., *Visualization of 3D Laser Scanner Data from an Archaeological Site Using OpenSceneGraph in ENEA-GRID Environment* ……..…… 295
D'Abate et al., *Laser Scanner Technologies and 3D Models Integration: Analysis and Methodological Approach*……..... 299
Agus M. et al., *Recent Results in Rendering Massive Models on Horizontal Parallax-Only Light Field Displays*……..... 304

EARTH SCIENCES

Coco S. et al., *VTA: Grid-Based Thermal Images Processing for Volcanic Activity Monitoring* 309
Marroccu M. et al., *Implementation of a Limited Area Ensemble Meteorological System in a Grid Environment* 315
Sannino G. et al., *Implementation of a Two-Way Grid Refinement Technique for the MITgcm: an Application to the Mediterranean Sea* 319
Reitano D et al., *The TEPHRA Model: Grid Implementation on PI2S2 Project* 327
Reitano D et al., *Monitoring and Analyzing of Volcanoes Activity by using INGV Cameras and Infrasound Sensors Network* 334
Sechi G.M. et al., *PON CyberSar Project Results on Drought Mitigation in Complex Water Systems* 340

ENGINEERING

Minutoli G. et al., *VisualGrid: Streaming and Visualization for Grid Applications Monitoring in gLite based Infrastructures*..... 348
Casula G.A. et al., *Automatic Antennas Design by means of Genetic Programming*..... 357
Coco S. et al., *Electromagnetic Field Strength Prediction in Urban Scenario by using a Grid Environment*..... 361
Catania V. et al.., *Multi-objective Design Space Exploration of Embedded Systems in a Grid Environment* 365
Bellavia F. et al., *A Non-Parametric Parallel Harris-Affine Detector* 372
Cipolla M. et al., *A Parallel Automatic Three-dimensional Fuzzy Edge Detector*..... 378
Manca A. et al., *A Multi-Airport Dynamic Network Flow Model with Capacity Uncertainty*..... 383

LIFE SCIENCES

Deias L. et al., *Synthesis of EBG Surfaces Using Evolutionary Optimization Algorithms*..... 391
Pappalardo F. et al., *Grid-based Atherosclerosis Simulations*..... 396
Pennisi M. et al., *A Biological Optimization Problem on the Grid*..... 403
Alemani D. et al., *Grid Based Cancer Growth Simulations*..... 411
Aprea G. et al., *Parameters Estimate in Metabolic Networks Reconstruction*..... 418
Licciardello G. et al., *PseudoBioRes: A New Comprehensive Database for Pseudomonas*..... 424
Boccia V. et al., *Recent advances of MedIGrid PSE in an LCG/gLite environment*..... 430
Mastriani E. et al., *PI2S2-ImmunoGrid Services Integration: a Web-Based Working Example*..... 438
Lombardo A. et al., *Performance of ViralPack applications on PI2S2 Grid infrastructure*..... 446
Di Grazia F. et al., *Brain Functional Synchronization on Grid*..... 452
Ardizzone V. et al., *Dose Distribution Verifications in Medical Physics: Results Obtained Using Geant4 and Grid Technology*..... 460

Pedretti A. et al., *Porting of a docking application on PI2S2 Grid infrastructure* 468

PHYSICAL SCIENCES

Muscato O. et al., *A Variance Reduction Scheme for Monte Carlo Semiconductor Simulations using the PI2S2 Grid*..... 474
Celino M. et al., *Materials for Hydrogen Storage: an ab-initio Molecular Dynamics Study in the ENEA-GRID Environment*..... 480
Chinnici M. et al., *Applications of Imaging Analysis to Tokamak Fusion Plasma by ENEA-GRID Technologies*..... 486
Chinnici M. et al., *Numerical and Statistical Tools for Image Analysis Based on the Database from Frascati Tokamak Upgrade*..... 490
Brescia M. et al., *The DAME/VO-Neural Infrastructure: an Integrated Data Mining System Support for the Science Commu*nity..... 496
Gambacurta D. et al., *Many-body systems HPC applications on the PI2S2 Grid*..... 506
Lanza A. F. et al., *A Grid-Based Comparison of Methods for the Detection of Earth-like PlanetaryTransits in the Time Series of CoRoT and Kepler*..... 513
Costa A. et al., *VisIVO: Data Visualization on the Grid*..... 518
Laurino O. et al., *A Web Application for Photometric Redshift Estimation*..... 524
Riccio G. et al., *Simulations of Cosmic Strings Signatures in the CMB*..... 531
Andreazza A. et al., *Overview of the Contributions of the LHC Experiments in INFN GRID for Bringing the Grid to Production Quality*..... 539
Carlino G. et al., *Distributed Analysis for the ATLAS Experiment in the S.Co.P.E Project*..... 547
Vuerli C. et al., *The Astronomy and Astrophysics Cluster in EGEE-III*..... 555
Castiglia G. et al., *Study of the Effects of Nuclear Motion on High Harmonic Generation in Simple Molecules*..... 563
Castiglia G. et al., *Three-Dimensional Numerical Model ofan* H_2^+ *Ion in Intense LaserFields*..... 569
Pasian F., *Data- and compute-intensive applications in Astrophysics: the Virtual Observatory*..... 577

The PI2S2 Project and the COMETA Consortium

Roberto Barbera[1,2,3], Marcello Iacono-Manno[2], Giuseppe Pappalardo[2,3]
[1]*INFN, Catania, Italy, roberto.barbera@ct.infn.it*
[2]*Consorzio COMETA, Catania, Italy, marcello.iacono@ct.infn.it*
[3]*University of Catania, Italy, pappalardo@dmi.unict.it*

***Abstract* — The COMETA Consortium was founded in 2005 by the three Sicilian universities, the main Research Organizations operating in Sicily, and the SCIRE consortium. COMETA's PI2S2 project was funded within the framework of a National Operational Program Call of the Italian Ministry of Education and Research. This paper describes the hardware and software facilities realized within the projects and reviews its main scientific results.**

***Index Terms*— COMETA, Computational Grid, Middleware, High Performance Computing, Modelling, Simulation.**

I. Introduction

A. The COMETA Consortium

THE COMETA Consortium [1] was founded in Catania, Sicily at the beginning of 2005 by the three Sicilian universities of Catania, Messina and Palermo, the Italian National Institutes for Nuclear Physics (INFN), Astrophysics (INAF) and Geophysics and Volcanology (INGV) and SCIRE, a public-private consortium based in Campania. Its members are:

1. INFN, the Italian National Institute for Astrophysics [2], active within the PI2S2 project with its Catania division;
2. INAF, the Italian National Institute for Astrophysics [2], active within the PI2S2 project with its Palermo and Catania Observatories;
3. INFN, the Italian National Institute of Nuclear Physics [3], active within the PI2S2 project with both its Catania division and Southern National Laboratories;
4. INGV, the Italian National Institute of Geophysics and Volcanology [4], active within the PI2S2 project with its Catania division;
5. the University of Catania [5];
6. the University of Messina [6];
7. the University of Palermo [7];
8. the SCIRE Consortium [8], formed by the University of Rome "Tor Vergata" and ELASIS, a private research agency founded by the FIAT group, and based in Pomigliano d'Arco, near Naples.

COMETA stands for "Consorzio Multi-Ente per la Promozione e l'Adozione di Tecnologie di Calcolo Avanzato", i.e. "Multi-institution Consortium for the Promotion and the Adoption of High Performance Computing Technologies".

The Consortium's goals are:

a) To promote the adoption and development of high performance computing technologies and solutions, with specific emphasis on the Grid paradigm, supporting large-scale modeling, simulation and data-management requirements of leading science and technology efforts within COMETA's founding institutions;
b) To bring innovation to the marketplace,

conducting client-sponsored research and development for public and private organizations;

c) To foster and promote higher education activities.

The COMETA Consortium's first initiative was to submit a proposal for MIUR's (the Italian Research and Education Ministry) call "Avviso 1575" [10] for research projects in the field of high performance computing and simulation. The proposal was indeed accepted and funded by the Italian Ministry of Education and Research with about 8.5 M€ for the three-year period 2006-2008. PI2S2 saw the participation of hundreds of researchers and technicians, whose aggregate manpower effort amounted to more than 1000 man months, partly hired directly by the COMETA Consortium and partly funded by its member institutions. During the three years of the PI2S2 project, the COMETA Virtual Organization (VO) has gathered hundreds of scientists and technicians from either inside and outside Sicily. Their interest in the exploitation of the Sicilian e-Infrastructure accounts for the more than 300 subscriptions to the COMETA VO as shown by Figure 1.

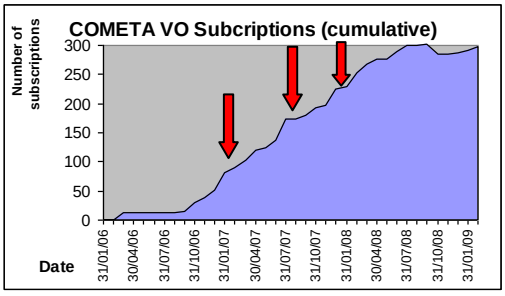


Figure 1: the growth of the subscriptions to the COMETA VO. The red arrows indicate some relevant training events that brought to massive increases.

B. The PI2S2 Project

The proposed project was nicknamed PI2S2 [10], the (Italian) acronym for "Project for the Implementation and Development of a Grid paradigm based e-Infrastructure in Sicily". The objectives planned and attained by the PI2S2 project are consistent with those of its submitting organization, i.e. COMETA, and financing institution, as stated in the "Avviso 1575" call. Specifically, PI2S2 was intended to:

a) Implement a Virtual Laboratory in Sicily, intended for both scientific and industrial applications;

b) Connect the existing and planned Sicilian e-Infrastructure to those operating at the Italian, European and world level, in order to favour outbound scientific cooperation on behalf of Sicilian e-Science and e-Industry, and increase its appeal and competitiveness;

c) Disseminate the Grid paradigm within Sicily's research, industrial and services communities;

d) Trigger and foster effectively the creation of spin-off companies in the field of Information and Communication Technology (ICT) area, so as to help tackling the endemic phenomenon of "brain drain" from southern to northern Italy and beyond.

In view of these strategic objectives, the PI2S2 initiative was conducted along four main lines:

a) Building high performance computing facilities across Sicily, to support the project's objectives;

b) Endowing the new hardware infrastructure with a suitable software kit, both at the middleware, the tool and the application level;

c) Ensuring a deep, wide and effective outreach of the project's facilities and results towards the Sicilian scientific and industrial communities, through a consistent series of well-publicized promotion events and training initiatives, uniformly distributed among all the project sites;

d) Enabling the development, by the re-

searchers involved and throughout the various participating entities, of a significant number of Grid-enabled, high-performance simulation and modelling applications.

Accordingly, in order to reflect the stated main project goals, as well as help managing its considerable scale and complexity, it was decided that PI2S2's activities be structured into four work-packages, defined as follows:

- WP1 – Design and Implementation of the Grid Infrastructure;
- WP2 – Middleware Integration with the Design and Implementation of Testing Procedures;
- WP3 – Design and Implementation of Hardware and Software Platforms for Training and Dissemination;
- WP4 – Application Development and Porting on the Grid (including Relevant High Level User Interfaces).

The rest of this paper is organized as follows. Section II describes the computing and storage facilities built within WP1; Section **Errore. L'origine riferimento non è stata trovata.** introduces PI2S2's research efforts pertaining to middleware/services and modeling/simulation, pursued by WP2 and WP4 and documented in Sections IIV and V respectively. The outreach and dissemination effort undertaken within WP3 is best documented through the Web [11], specifically by web pages announcing training events [12], conferences [13], other meetings [14] and initiatives (see, as an example, the "Researcher's Night" [15]).

II. PI2S2 Computing and Storage Facilities

The COMETA e-Infrastructure is a regional Grid platform both for scientific and commercial computation and storage. During the duration of the PI2S2 project it has been tuned to run heavily parallel jobs, thus combining High Throughput Computing (HTC) with High Performance Computing (HPC) paradigms into a single facility.

All installations are equipped with dual dual-core Opteron 2218, 64-bit, 2.6 GHz processors, with 2 MB L2 cache and 2 GB DDR2 RAM per core. They are arranged on a dual dual-core architecture with shared memory appearing to the user very similar to a quad-core system.

The sum over all the sites totals about 2000 cores, thus ranking the Sicilian e-Infrastructure among the most important Italian computing facilities.

All COMETA sites are equipped with an InfiniBand 4X Net Layer, capable of ensuring low-latency communication within a computing cluster. This technique results in a sharp performance improvement for High Performance Computing (HPC) applications, particularly when it is intended to run on several worker nodes. A Gigabit Ethernet connection is also available for Grid Services and trivially parallel (sequential) jobs.

The storage infrastructure is based on IBM DS 4200 Storage Systems that provide high features of redundancy, management and reliability. In fact, a DS 4200 Storage System supports several types of RAID and has an intrinsic redundancy of all critical components (fan, power, controller, etc.) to ensure maximum reliability; it allows expansions up to 56 TB each with SATA disks. Each Storage System is managed by two IBM x3655 servers that "export" the IBM GPFS parallel file system to all computing nodes.

The aggregate available storage capacity amounts to about 250 TeraBytes.

III. Introducing the PI2S2 Research Activity

Given the complexity and scale of the PI2S2 effort, planned project activities were organized into four work-packages, in accordance with the project architecture introduced in Section I.B. WP4, the "applications' work-package", focused most on research, gathering activities pertaining to a wide spectrum of disciplines in

the whole arena of science and technology. Further research was carried out within WP2, the "middleware work-package", in order to ensure, for the PI2S2 software development effort, wherever needed, innovative middleware functionalities or Grid-based services.

It is worthwhile to explain how the typical usage patterns arising in the Grid paradigm map to PI2S2's work-packages.

- *Loosely coupled concurrent computations*, especially *job farming* (consisting in a pool of sequential, independent, often replicated jobs) were addressed in WP4;
- *Parallel and distributed high-performance computation*, aimed at modeling tightly coupled, high-dimension systems, was also addressed within WP4. Effective execution of parallel, inter-communicating code was enabled by: (i) the InfiniBand high-performance, low-latency interconnection deployed by WP1, and (ii) the Grid middleware, MPI-related extensions developed within WP2;
- *Data intensive applications*, typically relying on and/or yielding huge, distributed datasets, account for a substantial share of WP4's activities. Some of them posed specific requirements (in terms of performance, security or ease of access), which were addressed within WP2, by suitable middleware extensions or customizations;
- *Collaborative scenarios* were systematically set up and exercised across all work-packages, for the making of scientific communities around the Sicilian Grid was the ultimate project goal;
- *Beyond batch-mode processing* Grid usage was pursued both through the implementation of portals for user-friendly job submission, and the development of Grid services and SOAs (Service Oriented Architecture) for the benefit of advanced application scenarios. In turn, such complex services required the introduction of feature-rich, value-added middleware, capable of guaranteeing an appropriate QoS (Quality of Service). This effort was therefore the realm of WP2, the "middleware work-package".

In this rich framework, the PI2S2's research activities flourished, eventually resulting in the conspicuous number of 122 applications officially supported in three years of the project. In this paper, general overviews will be provided in the following Sections IV and V, which deal with middleware- and modeling-related activities respectively

IV. Middleware and Software Services Development within PI2S2

Developing applications for the Grid, or porting existing ones to it, often implies a significant effort. In order to make application developers' life easier, PI2S2's WP2 has developed and deployed many tools to enhance the functionalities of the standard gLite middleware so as to simplify application integration for both parallel and sequential applications. The overall aim was to enhance flexibility, robustness, security in Grid resource access and management: this was in fact perceived as crucial for the Grid paradigm to gain acceptance among developers.

A. Job Control

GridFlex is a license manager that allows the safe delivery of licenses from COMETA license server to the remote sites over a public net; it is based on the FlexLM open source license manager adapted to the Grid environment. Specifically, the Grid authentication and authorization mechanisms have been integrated into the license server within a framework based on encrypted communication exchanges. Thus,

the user can be recognized, his/her authorization checked, and finally the file license delivered to the proper destination.

The WP2 team of middleware developers is currently extending the license server mechanism to cope with "*robot*" certificates as well. These represent a novel facility of many Certification Authorities, which allows a user to run a program on the Grid without any personal digital certificate. The user can store this special certificate on a smart card and insert it into a personal computer running the user interface (usually a web portal). This is enough to authenticate and launch a program on the Grid. This solution fits the requests from large user communities that want to simplify the approach to Grid computing.

The *accounting system* is another key element for a sustainable use of Grid infrastructures, especially when they are open to commercial exploitation. The gLite middleware offers various tools for resource metering. PI2S2 project developers extended the metering facilities to disk storage with a tool called SAGE.

The *Watchdog* utility consists in a couple of scripts that the user can integrate into his/her job in order to check the content of selected files at regular intervals. The mechanism is similar to the standard job perusal coming with the standard release of the middleware, but it offers two important additional features: (i) the interaction with the AMGA metadata catalogue permits several files to be saved automatically, producing the related metadata; (ii) the user can dynamically modify the list of saved files even based on the results of previous processing. There are other limitations affecting the file perusal technique in the case of parametric jobs. The Watchdog tool covers all kinds of jobs, as it consists of a couple of scripts that can be easily integrated into any types of job.

Granularity is a new tag added to JDL, the Job Description Language; which indicates how many cores must be reserved to a parallel job on the same physical processor. This parameter allows processing to be concentrated on the lowest attainable number of physical processors, so as to optimize performance of a set of parallel jobs by minimizing their mutual network (inter-processor) communication. This feature is particularly beneficial to shared memory applications that have been quickly (and maybe naively) adapted to the Grid distributed computing environment, and tend to perform poorly on it.

Finally, due attention has been paid to user-friendlier submission modes, capable of relieving the user of JDL scripting intricacies and approach the Science Gateway paradigm. *GENIUS* is a Web portal both for job submission and data management, which was also adapted to various applications, and for the integration of the above noted robot certificates. The Java-based environment *JGridCommunicator* addresses user-friendly job submission and graphical application management. The Grid2Win interface, developed as a joint Microsoft-COMETA effort, provides MS-Windows users with a familiar interface for accessing Grid services.

B. Support for parallel applications

Parallel MPI jobs need complex pre- and post-processing steps, which usually prepare the processing and collect the results, respectively. As many tasks are common to most jobs, some new wrappers have been developed and integrated into the middleware, and a middleware component called AMPIg: (Advanced MPI support) has been implemented. The user has only to select, in the submitted JDL file, the necessary tags to specify the selected compiler, the MPI library version and the chosen net layer. All the tasks previously carried out by the user are now accomplished by the new wrappers, including the file transfers from the master to the slave nodes. Thanks to the developed software, the user can now better focus on his/her application.

Transparent support for Grid deployment

was also the inspiration behind the DReflect effort which, even more ambitiously, purposed to offer automatic distribution over available worker nodes to standard, Grid-unaware, Java applications.

C. Novel Scheduling Policies

As a Grid platform runs many heterogeneous jobs, the scheduling policy has to be adapted to this variety. Ordinary policies privilege short jobs compared to long ones. This approach is too rigid for parallel HPC jobs, as it requires several cores at a time and would therefore remain waiting for a long time. Thus, HPC jobs have been separated from common sequential jobs by introducing a specific HPC queue. When a parallel job is queued, it reserves an adequate amount of cores. When enough cores are available, the parallel job can start. This mechanism is reasonably efficient if the number of short jobs overwhelms the others, so that turnover is fast enough. During reservation time, however, reserved cores can still be allocated to short jobs if the co-allocation policy is activated. In this case, after each short job has completed, the scheduler checks whether enough cores are now available for the waiting parallel job and starts it; otherwise, another short job is allowed to use the reserved cores for up to 15 minutes.

A few emergency jobs require an absolute priority as they deal with volcanic surveillance. The duration of such emergency jobs is short, and their frequency is rare, so the dedicated emergency queue is assigned a high priority and a pre-emptive scheduling policy. An incoming emergency job will preempt all the needed cores, sending the preempted jobs into a waiting state. Processing of emergency jobs will take no more than a few hours in the worst case, whereupon the previously preempted job is restored and restarted.

D. Data Management

Catalogue operations available with the standard middleware only cover single files. They do not include bulk (recursive) access to directories. The *lcg-recursive tools* allow code to up/download, to/from the catalogue, the whole content of a directory, descending the directory tree automatically, by a single command. A similar command is available for recursive deletion. These tools make the programming interface to the catalogue more powerful and versatile.

The availability of secure storage within an e-infrastructure is undoubtedly critical to gain industry acceptance for the Grid paradigm. The challenge is to store and maintain on a third party's Grid confidential and sensitive enterprise data, enabling secure and transparent access across the Internet, and, above all, protecting them against potential insider abuse.

As a solution, within PI2S2, the GS3 (Grid Secure Storage System) initiative exploited asymmetric (and symmetric) cryptography, to build a new layer providing encryption over standard Grid File Access Library (GFAL). The gLite-Security initiative also dealt with secure authentication for the PI2S2 Grid, based on X.509 and a RSA key pair stored on smart card.

E. Quality of Service on the Grid

The ability to guarantee a suitable quality of service (QoS) is paramount to secure acceptance of the Grid paradigm at the industrial level. Ideally, QoS assurance should take the form of actual *Service Level Agreements* (SLAs), describing precise constraints for specific QoS parameters such as response time, resource availability, throughput, bandwidth, etc. This is necessary to allow, in turn, application-level constraints, such as *deadlines*, to be fulfilled. While this would not necessarily be a requirement for the traditional e-science scenario, it would certainly apply to the e-business arena, and to a *user-centric* scenario, offering on-demand service provision to individual users. However, current SLA support is tailored to VOs rather than individual users' needs, which results in rather rigid, albeit

robust, resource management policies; these give control over priorities only, not scheduling policies or resource preemption, leading to "best effort" rather than guaranteed SLA support.

The aim of the SLAM (SLA Management) PI2S2 initiative was to reverse this situation, turning the Grid into a *service marketplace*, where individual consumers can dynamically negotiate separate SLAs, or reserve specific resources, with specific service providers. The resulting on-demand resource requirements can be satisfied through advance reservation of resources, which is the focus of the related ARM (Advance Reservation Management) initiative.

A similar approach was proposed by DynamicSLA.

Building upon gLite-oriented QoS management facilities, the S-Sicilia collaboration between the PI2S2 project and ORACLE aimed at implementing real-world business processes over the Grid. *Business SLA*s and suitable accounting and billing facilities were made available to support business-to-business relationships, as well as web application hosting.

V. Modelling, Simulation and Other Applications within PI2S2

The sheer number of applications developed within PI2S2's WP4 work-package, 122 as of time of writing, is a clear token of the effectiveness of the PI2S2 initiative, and of its significance and impact for the research community it was meant to serve and stimulate. Even more remarkable, perhaps, is the wide range of scientific disciplines and problem domains involved, spanning technology, and exact experimental and life sciences, and extending to humanities. Fields embraced are as diverse as mathematics, physics, astrophysics, engineering, chemistry, biomedicine, computer science, earth sciences, bioinformatics and even cultural heritage.

Other clear success indicators were the conspicuous growth rates observed for the number of the applications deployed on, and submitted to, the PI2S2 infrastructure, as well as the latter's resource usage figures. These provably demonstrate how, as PI2S2 developed, its HPC/Grid-based investigation model was progressively making inroads within the target research communities. A web portal [16] gives brief descriptions of the 122 applications whilst a comprehensive volume describing PI2S2 applications is also available [17]. Finally, present volume contains some papers on PI2S2 applications. Their overall distribution by domain is depicted by Figure 2.

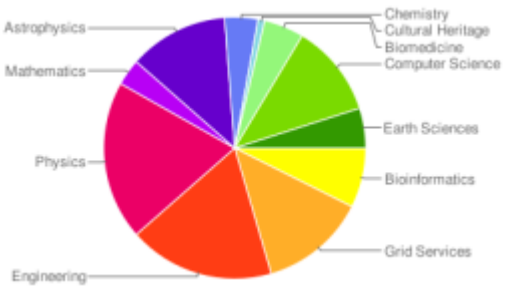


Figure 2: distribution of the 122 PI2S2 applications by scientific domain.

The rest of this section is an attempt at a concise, but comprehensive and systematic overview. For each application or homogeneous cluster thereof, we strive to identify the main themes addressed, together with the (computational) methods employed, and shed some light on the impact on the related field.

A. Astrophysics

Three main research lines in astrophysics can be identified for the PI2S2 project: virtual observatories, planetology, structure of the universe, jets and plasmas.

A virtual observatory is an ensemble of repositories of astronomical data, knit together by the Internet and suitable middleware, so that they can appear to researchers as a coherent

environment where meaningful virtual observations and experiments can be conducted, and scientific hypotheses validated. Grid naturally represents a very reasonable choice as the middleware underpinning a virtual observatory initiative. Research *themes* addressed include the analysis and classification of astronomical observations and cosmological data, and the simulation of atmospheric air showers. *Results* obtained pertain to: statistical data analysis and cross-correlations, simulations based on experimental data (which play a fundamental role in Astrophysics), facilities for the user-friendly visual presentation of massive data. Grid computing *methods* employed include recourse to the data storage facilities, and job farming (to ease solving N-body problems and applying MonteCarlo techniques). The COMETA Grid was successfully exploited by running the virtual observatory related applications CORSIKA, GEX, HEASOFT, MultiDIRECT and VisIVOServer.

In the field of planetology and structure of the universe, research *themes* addressed include analysis of light curves in the presence of planet transits, comet cloud evolution, galaxy dynamics and N-body simulation. *Results* obtained concern the origin of planet systems and the presence of extra-solar planets. The chief grid computing *method* employed is job farming (applied to the N-body problems and MonteCarlo techniques). Applications falling into this field were AstroDyn, COMP_FILT, FLY and ic_par.

Jets and plasmas were considered within several research *themes*, including: the magneto-hydrodynamics of astrophysical plasmas, jets in young stellar systems and active galactic nuclei, accretion phenomena in young stellar objects, supernova remnant shocks and nova explosions. *Results* obtained include detailed modelling of new satellite observations, fine diagnostics, the investigation of plasma effects unobtainable in a laboratory, geophysics-related simulations. Grid computing *methods* employed consist in the porting and execution of HPC parallel magneto-hydro-dynamics (MHD) modelling codes. Applications in this subject area were ADAP, GFLASH, JFLASH, P-Gadget2, PLUTO, MUSPH and ZEUSMP.

B. Bioinformatics

Within the realm of bioinformatics, two research lines were pursued: computational biology and genomics.

Computational biology *themes* investigated were the simulation of biological phenomena of relevance for human physiopathology, with special emphasis on models of atherogenesis and immune response to tumor vaccines. This has yielded *results* in terms of useful models of therapeutic treatments, and in silico drug screening, whereby later in vivo testing can be made more effective. Grid computing *methods* employed consist in the development and extensive execution of the parallel simulation codes embodied in the SimAthero and SimTriplex applications.

Genomics research *themes* addressed include the alignment of genomic, proteomic and nucleotide sequences, the classification and taxonomy of genetic sequence and structure information, the construction of phylogenetic trees. *Results* obtained provided an analysis of several instances of human, animal and vegetal genomes, and permitted some gene therapies to be investigated or tuned, and phylogenetic classifications to be achieved. Computing *methods* applied include alignment and search techniques and algorithms based on novel or non-standard concepts such as: Kolmogorov complexity, string theory, universal similarity measures and data compression, "k-nearest" classification, and (rather interestingly) genetic algorithms. The Grid infrastructure was used for job farming, for the sake of extensive testing, for building visual data presentation tools, and for parallel coding of advanced genome search techniques. The PI2S2 applications relevant to this area are BTR, ClustalW, Grid-K, KnetFold, MPGAfold,

SplitsTree and TOPALi.

C. Biomedicine

In the area of biomedicine and health care, a large number of efforts was pursued, yielding *results* that contributed to advancing practices and expertise within the local health care community, and embracing such *themes* as: stereotactic surgery, MRI and mammography image processing, digital archives for storage and management of DICOM medical images and other metadata, portals for medical images sharing within virtual organizations of medical doctors, researchers and students, integrated health care portals, oriented to long-term patient management by different health institutions.

Grid computing *methods* employed include: real-time parallel simulation (for the sake of Gamma knife surgery), massive preprocessing of medical images via job framing, data and metadata management/storage facilities for the implementation of health care repositories, and Grid-based collaboration platforms.

Applications pertaining to biomedicine are: BM Portal, DDSM_CONV, GammaKnifeRS, GRIDICOM, GridMRI, and MKDA.

D. Chemistry and Physical Chemistry

Themes treated in this area dealt with molecular dynamics (studying its models and the aggregation of complex molecules), condensation, catalytic surface reactions, and polymers (transport mechanisms and multiphase polymeric materials). *Results* were obtained improving our understanding of mechanisms underlying anomalous effects in hydrogen bonded liquids, the dynamics and energetics of interfaces between complex materials, and the properties and behavior of polymeric materials. Grid computing *methods* employed were job farming and the development, porting and/or execution of parallel, MPI-based codes. Applications relevant to the area at hand were: 1D_H2, 3D_H2+, BioSim4GRID, MCPolym, MolDyOnGRID, 3D-tdMC, DLPROTEIN, DESMOND, GAMESS, and SyMPoM.

E. Computer Science

The ability to guarantee a suitable quality of service (QoS) is paramount to secure acceptance of the Grid paradigm at the industrial level. Ideally, QoS assurance should take the form of actual *Service Level Agreements* (SLAs), describing precise constraints for specific QoS parameters such as response time, resource availability, throughput, bandwidth, etc. This is necessary to allow, in turn, application-level constraints, such as *deadlines*, to be fulfilled. While this would not necessarily be a requirement for the traditional e-Science scenario, it would certainly apply to the e-business arena, and to a *user-centric* scenario, offering on-demand service provision to individual users. However, current SLA support is tailored to VOs rather than individual users' needs, which results in rather rigid, albeit robust, resource management policies; these give control over priorities only, not scheduling policies or resource preemption, leading to "best effort" rather than guaranteed SLA support.

The aim of the SLAM (SLA Management) PI2S2 initiative was to reverse this situation, turning the Grid into a *service marketplace*, where individual consumers can dynamically negotiate separate SLAs, or reserve specific resources, with specific service providers. The resulting on-demand resource requirements can be satisfied through advance reservation of resources, which is the focus of the related ARM (Advance Reservation Management) initiative.

A similar approach was proposed by DynamicSLA. Building upon gLite-oriented QoS management facilities, the S-Sicilia collaboration between the PI2S2 project and ORACLE aimed at implementing real-world business processes over the Grid. *Business SLA*s and suitable accounting and billing facilities were made available to support business-to-business relationships, as well as

web application hosting.

F. Cultural Heritage

While a single application, ADAT, was developed in this area, a significant range of *results* was addressed, centering around the acquisition of ancient manuscripts of historical value, and including conservation, metadata-based semantic archival and classification, virtual restoration, Web- and Grid-based content provisioning for the implementation of digital libraries, intelligent character recognition (ICR), and, based on this, automated philological text analysis. Grid computing *methods* and facilities employed ranged from storage resources and metadata support, to fast parallel codes (including genetic algorithms) for fast ICR and text restoration.

G. Earth Sciences

Among the research *themes* investigated, significant effort was dedicated to volcanic and seismic activities monitoring, based on pattern analysis and recognition performed on visual and thermal images as well as infrasonic signals. Other activities of special interest for environmental safety were the modeling of lava streams forming on volcanic slopes and of the volcanic ash diffusion and transport in the presence of atmospheric phenomena (especially wind streams). *Results* of the noted research provided useful diagnostic methods and forecast models, intended for adoption by civil protection agencies for the sake of seismic and volcanic crisis management. Grid computing *methods* employed were job farming and the development and execution of parallel simulation code; data storage facilities were also exploited for managing massive geophysical experimental data collected from the INGV's sensor networks. Applications pertaining to this area are: C.I.E.E.V., Magflow, Meteo Station, MM5 model, P.R.E.S.A., TEPHRA, and V.T.A.

H. Engineering

In the area of engineering, three main research lines can be identified: mechanics and fluid-dynamics, electromagnetism, and systems and control.

Mechanics- and fluid-dynamics-related re–search *themes* investigated in PI2S2 were: flow models for gases and liquids, models of chemical and heat transfer phenomena in mechanics and fluid-dynamics technology (e.g. combustion), two-phase and multi-phase systems in the presence of external (mechanical or electromagnetic) forces. *Results* obtained were shown to have an impact on design practices for turbines and engines, shape profiles and edges (in naval/airplane construction), chemical reactors and clean room plants. Grid computing *methods employed* consisted in job farming (both sequential and mixed), parallel HPC, and Grid porting. Specifically, the well-known Fluent and OpenFOAM libraries were ported to the Grid so as to take advantage of its parallel computing facilities. Applications developed in this field were: B_nF, FDFP, GONAC, H2RECOVERY, HYBRID-BURNER, MoSChemPro, NSS, and PURGING-EFFICACY.

Electromagnetism *themes* pursued aimed essentially at modeling Travelling Wave Tube (TWT) devices, such as TWT electron guns and slow-wave structures, by means of the Finite Element Method (FEM). *Results* obtained find application in efficient design techniques for microwave devices, waveguides and TWT amplifiers. Grid computing *methods* employed include fast mesh generation and the development of parallel FEM solvers. The applications developed were COLLGUN and HELIX.

A third *theme* group, falling in the engineering area and tackled within PI2S2, pertains to systems and control. In particular, researchers investigated: the analysis of bio-medical signals, the simulation of complex and chaotic systems, natural language based

machine interaction, and mobile robots control. *Results* obtained improved our ability to design robots and other autonomous devices, and model social, economical and biological phenomena. Computing *methods* deployed on the project's infrastructure consisted in developing and running fast parallel codes intended for: time- and frequency-domain analysis, and control techniques based on soft-computing, game theory and Bayesian inference. The applications developed include: BioS, E^3, GNALIS, GoNet, and ICBANER.

I. Mathematics

Themes investigated in this area include: algebraic curves and coding theory; number theory; heat and current diffusion modeling, especially in semiconductors. *Results* obtained, by means of simulation runs on the PI2S2 infrastructure, include the validation of several conjectures and theories, and the derivation of novel semiconductor design techniques. Computational *methods* employed consisted in the development of parallel simulation codes and recourse to extensive job farming, especially for the sake of abstract space exploration. The applications developed in this area were: CR3x+1, GAG-codes, Grid_MC, and SUB_HEAT.

J. Physics

For mere convenience of exposition, PI2S2 physics activities can be broadly grouped into one of two areas: (i) structure of matter, and (ii) nuclear, atomic and cosmic ray physics.

In the first area, the following main research *themes* can be singled out: models of semiconductors and super-conductors, molecular ions in laser fields, meta-stability in long-range systems, nanostructures, quantum electronic phenomena. These led to useful *results* enabling: higher integration and the introduction of superconductivity features in electronic devices, improved production processes for microelectronic devices and nano-devices, nanotechnologies, laser processing of molecules, exploited (also) for ionization and dissociation. Computational *methods* employed consisted in Monte Carlo, job farming, as well as developing and/or running suitable numerical parallelizations; some effort went into implementing ordinary differential equation solvers by harnessing Matlab and job farming. PI2S2 applications pertaining to this field are: Abaqus/SIMULIA MSTJJ, nanotubi, PAMRED, REB, SicGrowD and SIMC.

In the area of atomic, nuclear and cosmic ray physics, the numerous *themes* investigated, embrace: multi-body systems at the atomic/molecular level, metal cluster many-body systems, excitation modes in nuclei, resonance phenomena, heavy ion reactions, neutrino and cosmic ray detection. *Results* obtained range from the validation of various theoretical aspects and hypotheses (e.g., unification, cosmic objects, the origin of the universe), based both on simulation campaigns and the analysis of massive experimental data, to the design of experimental and industrial-level devices and processes. Computational *methods* employed include Grid job farming, and the parallelization of codes from applications and standard libraries. The applications pertaining to this area were: ABINIT, Auger Offline, CACTUS, CMS, COMD, EXOTIC-MODES, Isospin, NEMO, PAR-EIGEN, Resonances ALICE, RPA-MBS, RSN package, SPEM, and VACE.

ACKNOWLEDGMENTS

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] COMETA Consortium, "COMETA Consortium Home Page", http://www.consorzio-cometa.it, 2009.

[2] Istituto Nazionale di AstroFisica (INAF), "INAF Home Page", http://www.inaf.it/ .

[3] Istituto Nazionale di Fisica Nucleare (INFN), "INFN Home Page", http://www.infn.it/ .

[4] Istituto Nazionale di Geofisica e Vulcanologia (INGV), "INGV Home Page", http://portale.ingv.it/

[5] Università degli Studi di Catania (UNICT), "UNICT Home Page", http://www.unict.it/ .

[6] Università degli Studi di Messina (UNIME), "UNIME Home Page", http://www.unime.it/

[7] Università degli Studi di Palermo (UNIPA), "UNIPA Home Page", http://portale.unipa.it/ .

[8] Consorzio di Ricerca per lo Sviluppo e l´Applicazione di Metodologie di Fluidodinamica Computazionale (SCIRE), "Consorzio SCIRE Home Page", http://www.consorzioscire.it/

[9] Ministero dell'Istruzione, dell'Università e della Ricerca. "Avviso pubblico n. 1575 del 29.11.2004 – Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione – Sistemi di calcolo e simulazione ad alte prestazioni", http://www.miur.it/UserFiles/1876.pdf.

[10] COMETA Consortium, "PI2S2 Project Web Page", http://www.pi2s2.it, 2009.

[11] COMETA Consortium, "Event Announcements", http://agenda.ct.infn.it/categoryDisplay.py?categId=14 , 2009.

[12] COMETA Consortium, "PI2S2 Training Events", http://agenda.ct.infn.it/categoryDisplay.py?categId=37 , 2009.

[13] COMETA Consortium, "PI2S2 Conferences", http://agenda.ct.infn.it/categoryDisplay.py?categId=18 , 2009.

[14] COMETA Consortium, "Other PI2S2 Meetings", http://agenda.ct.infn.it/categoryDisplay.py?categId=19 , 2009.

[15] COMETA Consortium, "La Notte dei Ricercatori", http://www.consorzio-cometa.it/nottedeiricercatori., 2008.

[16] COMETA Consortium, "Application Support Web Page", http://www.pi2s2.it/applications/, 2009.

[17] COMETA Consortium. "Proceedings of the "Grid Open Days at the University of Palermo", http://indico.ct.infn.it/conferenceDisplay.py?confId=24.

Roberto Barbera is a professor of Physics in the Faculty of Science at the University of Catania, Italy. His early research activity has been mainly in the field of experimental Nuclear and Particle Physics. He has been involved in many experiments on heavy ion collisions at intermediate energies, including the CERN LHC ALICE Experiment, for which he is the coordinator of the Off-line software of the Inner Tracking System. Since late 1999 he is interested in Grid Computing. He is a member of the Technical Board of the Italian INFN Grid Project, for which he coordinates both the GENIUS portal project and the GILDA virtual laboratory for Grid dissemination activities. At the European level, he is the coordinator of NA4 Generic Applications activities within the EGEE Project and has directed many training and dissemination events. Since 2001 he is also the member of the Technical Committee for the Grid Technology Area of the Trans-European Research and Education Networking Association (TERENA). He is the author of about 70 papers published in international peer-reviewed journals, and about 100 contributions to proceedings of international conferences. He is also a referee for the Journal of Grid Computing.

Marcello Iacono-Manno, after Physics graduation, worked on the Data Acquisition and Control Systems at the National Southern Laboratory of the National Institute for Nuclear Physics. He developed sequential and parallel readout programs and applications for electronic device control and used neural network software simulations to improve off-line data analysis. After his PhD, he joined the Grid team at the INFN Catania Section, collaborating with the Trigrid and PI2S2 projects as an application supporter, focusing on parallel and massive calculus. Within these projects, he has also organized many events for training and dissemination. He has published several papers on international journals on either his former and current research activities.

Giuseppe Pappalardo is a professor of Computer science in the Faculty of Science at the University of Catania, Italy. His research interests are in the field of distributed systems and software architectures, focusing on large scale distributed systems and the Grid, with special emphasis on resource allocation, concern separation in object-oriented design, and formal verification techniques. He has authored over eighty scientific articles and chaired international conferences on emerging Grid technologies and the separation of design concerns. He has been involved in national and European projects on computer networks and the computational Grid.

MAGO
Monitoring All Grid Objects

Carmine Spizuoco[1], Francesco Adinolfi[1], Giovanni Bracco[2]

[1]*Consorzio Campano per l'Informatica e l'Automazione Industriale (C.R.I.A.I.), Piazzale E. Fermi 1, Loc. Porto del Granatello, 80055 Portici, Napoli, Italia.*
Phone:+39-081-776-6905, Fax:+39-081-776-0583
c.spizuoco@criai.it
[2]*ENEA FIM Via E. Fermi 45, I-00044 Frascati (Roma) Italy*

***Abstract* – This paper describes a monitoring system, MAGO (Monitoring All Grid Object)[1], designed in the framework of the research activities for innovative solutions in distributed and Grid computing, inside the CRESCO Project (Centro computazionale di RicErca sui Sistemi COmplessi)[2].**
The system has been designed and developed at the CRIAI lab [3], an ICT industrial research centre, in Portici (Naples), in collaboration with ENEA.
The first step of research activity was an exploration of several available Open Source monitoring systems which identified Ganglia as the starting point solution for the development.
Ganglia [4] is a system management tool allowing distributed monitoring and permits to control group of computers managing multiple parameters as: processor load, disk space, memory usage.
MAGO allows the integration of Ganglia default information schema with many different data that are specific to the ENEA-GRID [5] infrastructure: demon processes, services and software applications. The collected information are centralized in a resource data repository, based on MySql.
The MAGO system can be considered as a "case study" for a software architecture where open source tools are integrated and tailored to fulfil the requirements specific for a given environment as the monitoring of ENEA-GRID infrastructure.
The implementation makes use of the Andrew File System, an open source component of ENEA-GRID, to distribute the near-static configuration information about the required monitoring of the infrastructure nodes. This choice does not limits the generality of the proposed solution, as the role of OpenAFS as information distribution tool is clearly identified so that other information distribution methods could be in principle introduced without loosing the overall architecture design. The system includes also an alert system based on a 'hard real time' feature using low-level system calls to provide high performance alert management.
Our goal is to develop an advanced monitoring system, which allows choosing run time metrics relevant to a particular group of hosts belonging to ENEA GRID.

***Index Terms*— Monitoring Resources, Grid Computing, Glue Schema, Ganglia, Nagios, UDF, AFS, XML, RRDtool, XDR, CRESCO, Grid**

I. INTRODUCTION

Over the last ten years, there has been an enormous shift in high performance computing from systems composed of small numbers of computationally massive devices to systems composed of large numbers of commodity components.
This architectural shift from the few to the many is causing designers of high performance systems to revisit numerous design issues and

assumptions pertaining to scale, reliability, heterogeneity, manageability, and system evolution over time.
One of the key challenges faced by high performance distributed systems is scalable monitoring of the system state. Given a large enough collection of nodes and the associated computational, I/O, and network demands placed on them by applications, failures in large-scale systems become commonplace.
There are many factors that can cause troublesome situations in distributed systems. For these reasons it is important to realize a model to discover the anomaly of system, this process is called "*monitoring*"**.**

II. SCOUTING

After the review of the available open source monitoring tools, Ganglia and Nagios [8] result as the two most interesting solutions.
Nagios is a system and network monitoring application. It watches selected hosts and services providing alert messages.
It was originally designed to run under Linux, although it should work under most other platforms.
Some of the many features of Nagios include:

- Monitoring of network services
- Monitoring of host resources (processor load, disk usage, etc.)
- Simple plug-in design that allows users to easily develop their own service checks
- Parallelized service checks
- Ability to define network host hierarchy using "parent" hosts, allowing detection of and distinction between hosts that are down and those that are unreachable
- Ability to define event handlers to be run during service or host events for proactive problem resolution
- Automatic log file rotation.

Ganglia is an open-source project that grew out of the University of California, Berkeley and it is a scalable distributed monitoring system for high-performance or/and high-throughput computing infrastructures such as clusters and Grids. It is based on a hierarchical design targeted at federations of clusters.
It leverages widely used technologies such as:

- XML for data representation
- XDR for compact, portable data transport
- RRDtool for data storage and visualization and provides the following functionalities:
- Monitoring of host resources (processor load, disk usage, etc.)
- Ability to define subnet host using the udp communication protocol and the multicast groups
- Graphical User Interface to show the results
- Communication based on request/response model.

It uses carefully engineered data structures and algorithms to achieve very low per-node overheads and high concurrency. The implementation is robust, has been ported to an extensive set of operating systems and processor architectures, and is currently in use on thousands of clusters around the world. It has been used also to link clusters across university campuses and can scale to handle clusters with 2000 nodes.
In order to make a careful analysis about the implementation of a monitoring system compliant to the infrastructure ENEA-GRID, CRIAI has emulated the characteristics of the hosts belonging to ENEA, to verify the behaviour of the monitoring tools.
An analysis of software shows that the Ganglia monitoring tool is the most suitable for our needs for several reasons:

- It is based on mature technologies: XML XDR RRDtools
- The connection of the multicast group, allows to know the status of the subnet with a single request to a single host
- The addition of new metrics in the default

metric is simple and inexpensive.
The above statements refer to version 3.0.6 (Foss) [9] of Ganglia issued in December 2007.

III.ARCHITECTURE

MAGO allows the integration of Ganglia default information schema with many different data that are specific to the ENEA-GRID infrastructure: demon processes, services and software application.
As shown in Figure 1, the ENEA-GRID infrastructure is based on mature technologies such as:

- A Graphic user interface based on Java and Citrix technologies
- LSF Multicluster [7] a resource manager
- AFS[6] a distributed file system

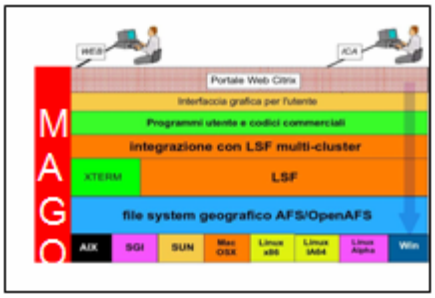


Figure 1 Infrastructure ENEA-GRID

In this context MAGO is interfaced for monitoring purposes to all the levels of ENEA-GRID components.
In this work it was chosen to create a hierarchical structure, as shown in Figure 2. One of the peculiarities of MAGO is the capability to be centralized, allowing availability and robustness, also in case of crash involving specific computers.
In detail the system bottom-up architecture is based on three layers. The first is called "*node*". It represents the single machine; the second level is called "*Site Server*". It is composed by servers geographical located in ENEA sites (e.g. Bologna, Portici, Frascati, etc).

The third level is called "*Master Server*" .It is a central server observing and controlling the entire network of ENEA-GRID.

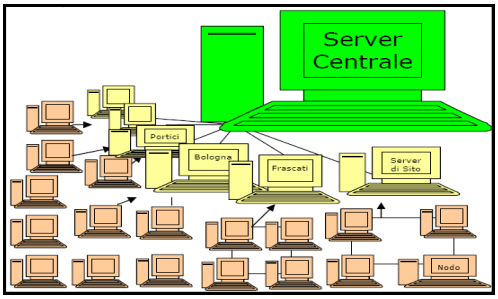


Figure 2 Level view of MAGO

MAGO uses a combination of different technologies:

- XML[10]
- Ajax [12]
- J2ee[16]
- Struts[11]
- UDF Mysql[14]
- Bash Shell Scripting[13]
- Application Server JBOSS[15]

The usage of these different technologies made it possible to implement a flexible, modular, scalable and fault tolerance monitoring tool.
Figure 3 shows the interactions among the main components of the system.

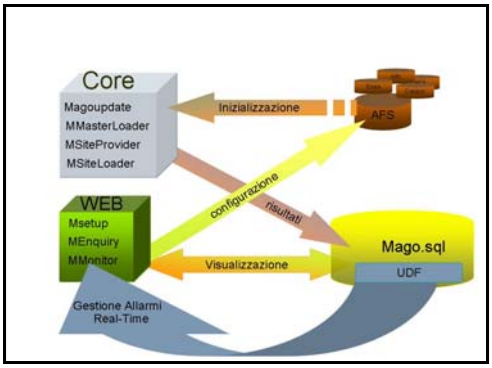


Figure 3 Interaction between components of MAGO

1) MAGO Web

MAGO Web Application is the user interface component. It provides several features:

- Access to the system through a secure connection on a CRESCO front-end host, compliant to ENEA-Grid security
- Configuration of each subsystem:
  - Host
  - Site
  - Metric
  - Alarm
- Display results

2) AFS

Andrew File System is a distributed filesystem. It offers a client-server architecture for federated file sharing and replicated read-only content distribution, providing location independence, scalability, security, and transparent migration capabilities.
The implementation of MAGO makes use of this open source component of ENEA-Grid, to distribute the near-static configuration information about the required monitoring of the infrastructure nodes.

3) MAGO CORE

This component is composed by 4 elements:

- **MMasterLoader**
  It resides on the Main Server, synchronizes parses and saves all the data in the central database.
- **MSitLoader**
  It resides on the Site Server, synchronizes parses and saves all data at site level.
- **MSiteProvider**
  It resides on the Site Server and retrieves information from Ganglia daemons.
- **Magoupdate**
  It resides in AFS and it is the script managing installation, configuration and update of the hosts to be monitored.

Figure 4 shows the information path from a Site Server (in one of ENEA site) to the Main Server at CRESCO site, in Portici.

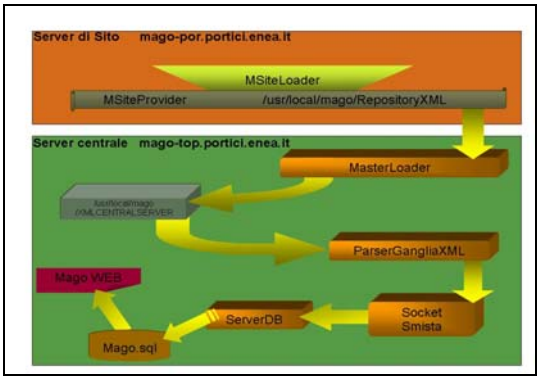


Figure 4 Information path

The highly modular design of MAGO made it possible to realize a fault management so that even if one process goes down all process restart, as shown in Figure 5

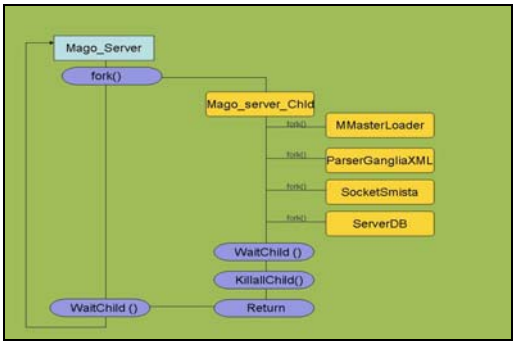


Figure 5 Fault tolerance module

4) MAGO DATABASE

It's located on the central server, under the name of Mago.sql.
It contains:

- Sites and hosts to be monitored
- List of metrics added to Ganglia
- Lists of multicast groups

- Monitoring results

It is also involved in the alert generation by taking advantage of the "User Defined Function", available starting from version 5.0 of MySQL.

IV. CONCLUSION

This work adds new functionalities to the management of ENEA-GRID Infrastructure by extending the capabilities of Ganglia.

It allows monitoring with standard credentials of ENEA-GRID. It allows users to check the performance of the whole system and of each individual resource. It allows the addition of dynamic dedicated metric. A historic central database has been realized for easy consultation. A fault tolerance model has been realized.

Finally, this work creates and implements functionalities not supported by the version of Ganglia available at the project time. Subsequent Ganglia releases have addressed the same issues.

REFERENCES

[1] http://www.cresco.enea.it/LA1/cresco_sp12_criai/
[2] http://www.cresco.enea.it/
[3] http://www.criai.it/
[4] http://ganglia.info/
[5] http://www.eneagrid.enea.it/
[6] http://www.afs.enea.it/
[7] http://www.platform.com/
[8] http://www.nagios.org/
[9] http://ganglia.info/?p=60
[10] http://www.w3.org/XML/
[11] http://struts.apache.org/
[12] http://www.asp.net/ajax/
[13] http://www.gnu.org/software/bash/
[14] http://www.mysqludf.org/
[15] http://www.jboss.com/
[16] http://www.j2ee.it/

ACKNOWLEDGMENTS

The authors would like to thank:
Vincenzo Masucci, Anna Jannace, Pasquale Sannino, Luigi Frappola, Michele Masullo

Carmine Spizuoco
C.R.I.A.I. (Consorzio Campano di Ricerca per l'Informatica e l'Automazione Industriale)
Piazzale E. Fermi - Loc. Granatello
80055 - Portici (NA)
www.criai.it
e-mail: c.spizuoco@criai.it
carmine.spizuoco@criai.com

The S.Co.P.E. Project

L. Merola[1,2]
[1]*Università degli Studi di Napoli Federico II, Napoli, Italy, leonardo.merola@na.infn.it*
[2] *on behalf of the SCoPE project*

Abstract— **This paper describes the project SCoPE, which started at the beginning of the year 2006 and was completed in three years, with the construction and activation of a Data Center which is now perfectly integrated in the national and international Grid initiatives. The DataCenter hosts about 300 eight-core blade servers, 220 Terabyte of storage, and is already able to accommodate 500 more servers. The scientific applications which the Data Center is used for, in the areas of Astrophysics, Chemistry, Mathematics, Medicine, Engineering, Physics, are briefly described.**

***Index Terms*— SCoPE, Gric, scientific computing.**

I. INTRODUCTION

THE SCoPE project started in the year 2004 from an idea of a few researchers at the University of Naples *Federico II*, Italy, who recognized the need for a scientific Data Center, to be used in the framework of a more general Grid infrastructure. The acronym stands for "*Sistema Cooperativo per Elaborazioni Scientifiche Multidisciplinari*", that is a collaborative system for scientific applications in many areas of research. At the end of 2004 the group decided to submit a request for funding to the *Ministery of University and Research*, and the proposal, which gathered more than a hundred researchers from the University of Naples, was submitted in February 2005, receiving its final approval exactly a year later, in February 2006, when all activities started.

II. THE INFRASTRUCTURE

A. The need for a new infrastructure

THE University of Naples is a very large one, with more than 100,000 students and more than 3,000 professors and researchers. However, no significant scientific Data Center existed before the SCoPE project and most of the Departments in the Faculties of Sciences and of Engineering had their own small centers, with dedicated infrastructures.

With the SCoPE project, the proposing group, which had grown up to 128 people, agreed to build a new, single Data Center, in the Monte S. Angelo Campus, which already hosts the Faculty of Sciences and is very near to the campus of the Faculty of Engineering. Excellent network infrastructures were already operational, with many kilometers of optical fibers, structured according to a multi-ring shape with multiple differentiated ways connecting all the departments; each fiber is operated at 1 Gbps or 2.5 Gbps, with several fiber pairs on each connection. The network infrastructure of the University *Federico II* is shown in the map below.

B. The design of the Data Center

The design of the Data Center was completed in a few months, according to the following constraints:

- Localization in the Monte S. Angelo Campus, in a small new building, max 150 square meters;
- New power plant, capable of delivering 1 MW of electric power in a continuous mode;
- Efficient cooling system, capable of dissipating 2 kW per cubic meter, and 30 kW per rack;
- Standard (Gigabit Ethernet) networking infrastructure, with a high capacity switching fabric;
- Low latency (Infiniband) networking infrastructure, with a single switching fabric for each group of 256 servers;
- Large storage capacity, both NAS (Network Attached Storage) working with the iSCSI protocol, and SAN (Storage Area Network), working with a Fibre Channel Infrastructure;
- Open Source (Scientific Linux) for the operating system;
- Integrated monitoring system for all the devices of the Data Center, able to monitor the most relevant parameters of server, storage, networking, as well as all the environmental parameters (temperature, humidity, power consumption etc.)

C. The realization of the Data Center

The following image shows the Data Center whose construction started at the end of 2007, after two European tender procedures were completed.

The power plant, which is shown below, has three UPS's for a total of 1 MW and a diesel power generator with the same power, supplied with 3,000 liters of gasoline.

A special effort was devoted to the cooling system, made by RITTAL: it is a liquid cooling system, with two redundant chillers (below, left) and a distribution system (below, right) which delivers cool water to each rack, equipped with redundants LCP units and brings back the heated water to the chillers.

The interior of the Data Center accomodates as many as 33 racks, leaving plenty of space space for mantainance operations.

The racks are organized in three groups of "10+1", where the "1" is used for passive cabling, and the "10" are used for switchs, servers and storage.

All the hardware in the Data Center was given to a single general contractor, which was DELL. They provided the servers, the networking switches, the storage, the monitoring system.

D. The servers in the Data Center

The servers are basically all blade servers, arranged in groups of 16 blade per chassis, as shown below; each blade has 2 quadcore CPU's by Intel, 8 or 16 Gbyte of memory per blade. The chassis has pass-through switches for Gigabit Ethernet (copper) and for Infiniband (fibre).

E. The network in the Data Center

The main gigabit network is based on Cisco 6513 switches. Each pairs of them provides full-wire-speed connectivity to all blade servers in the "10+1" group of racks, in a fully redundant architecture which guarantees that the switching matrix is unique for the whole group of servers, avoiding bottlenecks in transferring data from one switch to another. The switches, shown below, are however interconnected each other with a pair of 10 GbE lines for each switch.

The same concept is applied for the Infiniband switches, shown below, which are also made by Cisco. We used Fibre connections instead of the classical coax cable of Infiniband, to reduce the size of cabling and to increase in realiability because of the lack of interferences.

The passive cabling was rather complicated, because of the thousands of cables which had to be deployed in the Data Center (see below for copper and fibre, respectively); most of the cabling is on aereal guides, and not under the pavement, in order to ease maintenance.

F. The storage in the Data Center

The storage system, shown below, is based on EMC^2 hardware, and in particular on the CX3-40 SAN system with 130 TB of storage, with a Fibre Channel infrastructure which connects each blade chassis, but we also use a couple of EqualLogic NAS (see below), based entirely on the iSCSI protocol, and with capacity of 16 TB for each system.

G. The Control Room

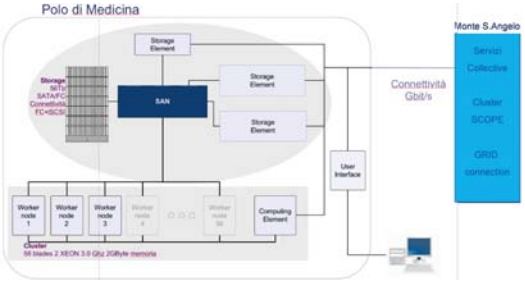
The small size of the Data Center (less than 150 square meters) does not allow for human presence inside; therefore we used the nearby open space, shown below, to accommodate all the operational people, involved in monitoring, troubleshooting, maintenance and software development. More than 20 desktop PCs are connected via a dedicated (and redundant) fibre channel conncetion to the Data Center, which is however only 50 meters away. Large-size (58 inches) flat panel displays show in real time the status of the Data Center.

H. The remote sites

Three satellite installations ("remote sites") are present in the SCoPE infrastructure, namely at the Medicine Faculty (*site I*), at the Chemistry Dept. (*site II*) and at Applied Mathematics Dept. (*site III*). *Site I* hosts a 50 TB storage system, which is connected to the SCoPE Data Center through a 5 km fiber optic link, using the fiber channel protocol, as well as through a similar link transporting TCP/IP at 2.4 Gbps (packet over Sonet). *Site II* hosts a rack with 4 multi-processor servers, 32 cores per server, 2 GB of memory per core, and is just 200 *m* away from the Data Center, to which is connected via 2 optical links operating at 1 Gb/s. *Site III* hosts a rack with 16 blade, 8 core per blade, 2 GB of memory per core; also *site III* is just 200 *m* away from the Data Center and is connected in the same way.

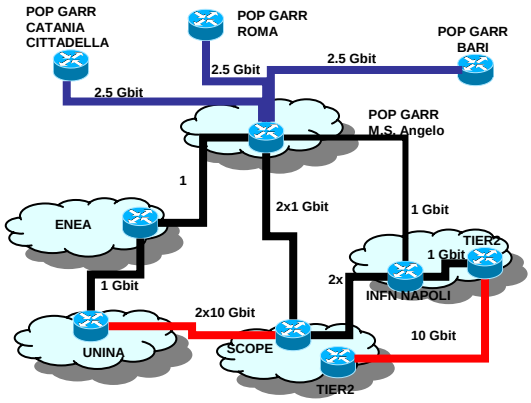
In addition the SCoPE Data Center is connected at 10 Gbps to the INFN ATLAS Data Center which is located very close to it, inside the Physics Department building, which also hosts the Napoli Section of the INFN.

As an example of remote site, see the figure below, which is located within the area occupied by the Faculties of Medicine and Biotechnology. A 100 CPUs cluster provides computing resources mainly dedicated to support interactive computational requests in the field of bioinformatics, but its nodes are also connected to the university Grid through a local computing element that grants access to the nodes. The storage system based on EMC2 hardware, and in particular on theCX3-40 SAN system, consists of 60 TB dedicated to biological databases and bioinformatics tools. The storage system is directly connected to the local nodes, but also, via dedicated optical fibres, to the main plant nodes, which have fast access to the stored data. A custom developed fast scheduler takes care of distributing the computational requests to the local nodes, for interactive tasks or to the remote nodes, for less immediate, more complex and heavier jobs.



I. The wide area network connections

Starting from the Monte S. Angelo network exchange center, hosting the GARR (Italian NREN) GigaPOP the SCoPE Data Center is connected to the outside networks through 3 POS STM-16 (2.5 Gbps) fiber circuits reaching respectively the Roma, Catania and Bari GARR GigaPOPs. It is also connected with 2 10 Gbps Ethernet direct connections to the Napoli Metropolitan research Network, offering Giga-speed access to all the Universities and research institutes located in the Napoli urban Area (INFN, CNR, Enea, Unina, Orientale, Parthenope, SUN etc.).

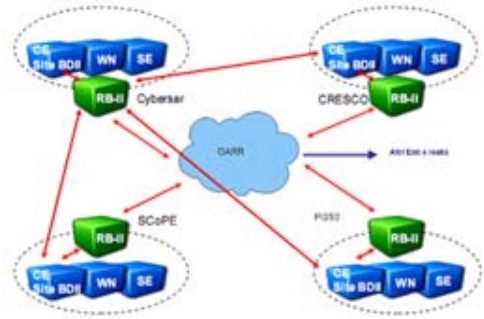


J. Interoperability

The Italian Ministry of University and Research has promoted the creation of an unified e-infrastructure and a new scientific-technical community [1], in the South of Italy by the synergy among the four e-infrastructures: CRESCO [2], CyberSar,[3] PI2S2 [4] and SCoPE. These projects have experienced a very successful cooperation

since 2007, with the aim of sharing resources and knowledge, through a new interoperability model.

From a technical point of view, this approach provides an integrated production-quality service on four different e-Infrastructures based on the gLite middleware and the ENEA-Grid experience. Each e-infrastructure deploys a minimal subset of gLite collective services, called Primary Services (as shown in the picture below), in order to manage the access to resources and the membership of the authorized virtual organizations. This set of collective services, duplicated at each project infrastructure, avoids single points of failure.



This interregional experience, has been the basis for the new GRISU' e-infrastructure [5], which plans to extend the above interoperability model to other partners in the South of Italy, such as SPACI [6], and to be part of the planned Italian Grid Infrastructure (IGI) within the European Grid Infrastructure (EGI).

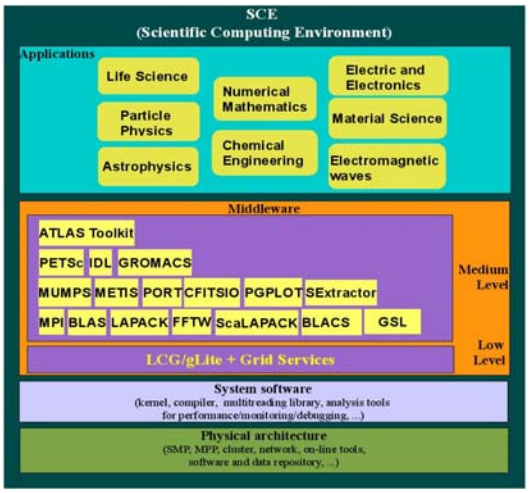
The new interoperable GRISU' Grid infrastructure provides more than 10.000 logical CPUs, 500 TB of raw disk to support a large scientific community with hundreds of applications. GRISU' is now ready to export this experience in other national and international research and industrial frameworks.

III. THE SOFTWARE

THE SCoPE Grid solution is based on the INFN-GRID release, realizing a structured architecture built on a customization of the LCG storage and computing infrastructure projected for the high-energy physics community using the Large Hadron Collider (LHC) and the gLite middleware developed within the EGEE project context. It offers a computational and storage environment where the Resource Framework Layer (RFL), the Information System Framework (IS) and the Information Data Model encompass semantic and interaction models for all the available resources. The actual implementation provides the creation of a central Virtual Organization (VO) named *unina.it* managed by a local Virtual Organization Membership Service (VOMS) which collects all the users of the project and is enabled in each shared resource. Other VOs, are enabled respectively on the resources used in explicit local groups associated to specific experiments. The

centralized collective services offer Services Discovery, Resource Brokering, File Catalog, Grid Monitoring and graphical user interface facilities. The site gLite component provides two Computing Element (CE) in High availability, batch execution systems, with 304 associated Worker Nodes (WN) offering computing power, together with 10 Storage Element (SE) servers based on the Disk Pool Manager (DPM).

A *localized* medium level middleware, named SCOPE Toolkit, including all the mathematical packages, software and libraries needed to the Unina researchers, has been designed and realized [7]. The architecture of this scientific computing environment is depicted below.



For what concerns the operating system, all machines use the freely available Scientific Linux 5.2; in general, only public domain software is used in the project, apart from a few commercial applications, like IDL from Research Systems, inc.

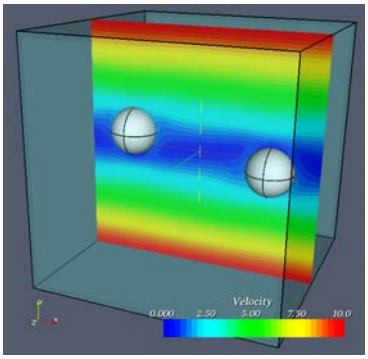
The monitoring system is based on proprietary, but free, tools from the vendors (DELL, RITTAL, CISCO), and on NAGIOS for the network monitoring. A higher level monitoring is provided by the gridICE utility.

IV. The Scientific Activities

A. *The area of material sciences*

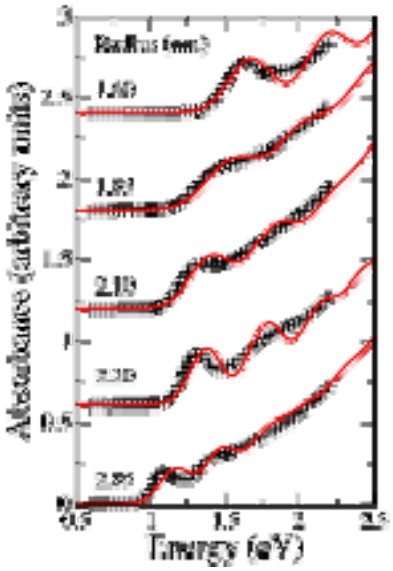
Several applications have been studied within SCoPE; among these, very promising is the study of the behavior of suspended particles in liquids. Suspensions of particles in liquids are a class of materials relevant in a huge variety of applications, e.g.: i) polymer melts with fillers; ii) biomedical materials; iii) food; iv) cosmetics; v) detergents. This variegate spectrum is due to the differences in particle concentration, mechanical properties, shape and size. Particles suspended in viscoelastic media are known to develop structures when sheared at sufficiently high shear rates. Jefri and Zahed [8] experimentally analyzed suspensions in planar Poiseuille flow, with the following three results: a) in Newtonian suspensions the particles were uniformly distributed in both the transverse and axial directions; b) in the shear thinning fluid migration of particle toward the upper and lower plates was observed followed by string formation close to walls; c) in the constant viscosity elastic fluid the particles were found along the tube axis with very few particles on the upper and lower plates.

The aim of these studies in the SCoPE project is to characterize the flow behavior of dilute and semidilute suspensions of rigid spheres in viscoelastic media in confined geometries (e.g. [9-10]). We considered: i) dilute and semidilute suspensions; ii) rigid; iii) spherical; iv) nonBrownian inclusions; v) confined geometries. We used intensive numerical simulations in 2D and 3D FEM. This code uses the following libraries: a) BLAS/LAPACK; b) METIS. The code is in Fortran 95, and the SCoPE group used the Intel Fortran Compiler v 9.1.043 and the G95 Fortran Compiler. Moreover, the following solvers were used: a) HSL library; b) MKL Gmres; c) Sparsekit; d) Pardiso; e) Mumps; f) Petsc. PARDISO gave the best results in terms of efficiency. A simulation is given below [11].

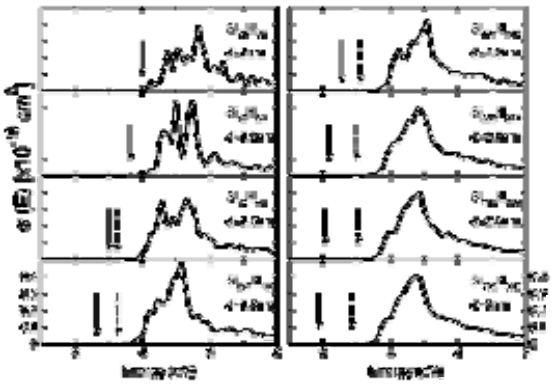


Another research area is related to emerging fields of nanotechnology which is believed to be the basis for innovative materials and devices in the near future. In a nutshell, Nanotechnology is the ability to engineer materials and devices on a length scale as small as several nanometers. The optical and transport properties of such nanostructures cannot be described in terms of macroscopic parameters and therefore a microscopic or atomistic viewpoint is called for. There is the need of developing computational methods that, given the complex nature of a nanostructure, must be tailored on massively parallel supercomputers in order to reduce the code execution time. In this respect, the SCOPE project has offered a tremendous opportunity for studying the optical properties of large Silicon and InAs nanocrystals. It has been, indeed, developed an efficient computer code based on the tight-binding method which is able to work out both the nanocrystal electronic structure and the optical absorption spectrum $\alpha(\omega)$ using two level of description. In the first level, the polarization induced on the nanocrystal surface by the external field is ignored (this approximation is known as RPA) while in the second this surface contribution is fully taken into account. Comparing the numerical results with experimental data it turned out that for a nanocrystal the surface polarization plays a fundamental role while, for a bulk material, it only gives a marginal contribution. In the following figure a direct comparison between numerical [12,13] (red lines) and experimental (crosses) data [14] is shown for the case of colloidal InAs nanocrystals with a spherical shape and with different dimensions. As it is seen, the agreement is very good.

An important conceptual point in the physics of semiconductor nanocrystals is the evolution of the optical response with the dimension. Since the number of atoms making the structure increases very rapidly with the nanocrystal diameter, these studies require significant computational resources. In the other figure, the absorption cross section of hydrogenated spherical silicon nanocrystals is shown as a function of their diameter within the RPA approximation [15].



The two arrows in each panel indicate the HOMO-LUMO (full line) gap and the absorption onset (dashed line). The important piece of information coming from that figure is that the absorption evolves rapidly to that of bulk silicon. About 700 Silicon atoms (2 nm in diameter) are almost enough for reproducing the typical bulk silicon absorption. This finding is relevant for Silicon optoelectronic applications.

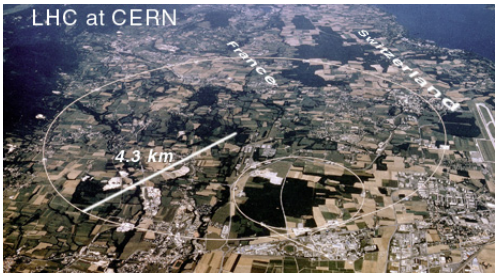


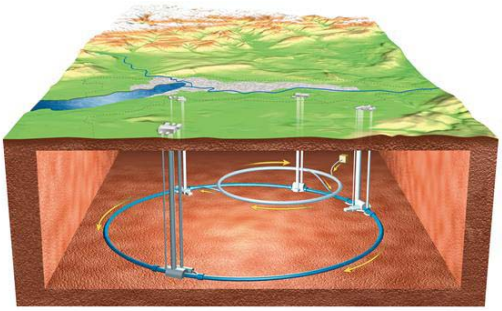
There are some other areas which have been touched during the SCOPE project; it is worth mentioning the problem of screening studied with both the tight-binding and ab initio methods [13,16,17], the role of oxygen vacancies on SnO_2 nanobelts [18] and some questions related to the physics of carbon nanotubes [19]. In the latter case the work has been done in collaboration with STMicroelectronics.

B. The area of particle physics

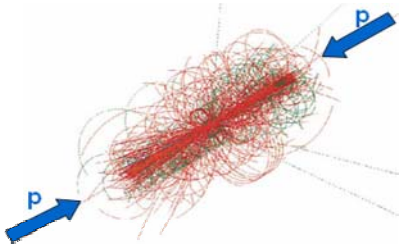
Nowadays, most of the particle physics activities are devoted to the search for the Higgs boson. Higgs boson is the particle whose existence has been postulated [20-22] by the Scottish physicist Higgs, to explain the difference in mass between *photons* and *vector boson W* and *Z*; Higgs boson should interact with all other particles, thus fixing their mass value.

The Large Hadron Collider (LHC) at CERN [23] (photos below) will allow collisions at very high energies, between protons and protons, thus freeing energy that will create other particles, among which physicists expect to reveal also the Higgs boson.



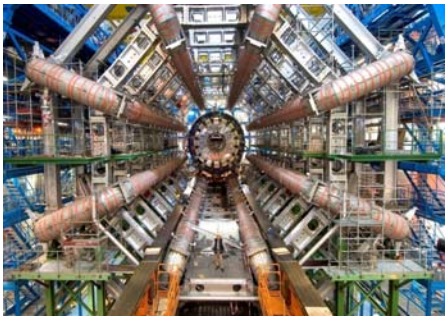


This particle cannot be seen directly, but through the decay products, such as 4 high-energy muons (see below).

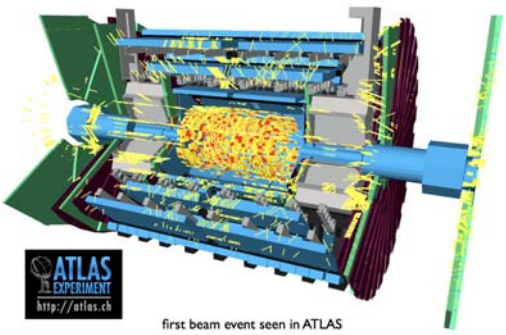


The scientific analysis of the huge amount of LHC data is possible only in Data Centers which adopt the computational Grid model [24,25], but a great effort is still needed to optimize the use of the systems and to completely analyse the data.

ATLAS [26], whose main apparatus is shown below, is one of the two main experiments at LHC. Naples, and the SCoPE project, has its role in ATLAS, as a Tier-2 site [27] for the experiment.



A reconstruction of the first beam event seen in ATLAS in September 2008 is shown below.



C. The area of life sciences [28-30]

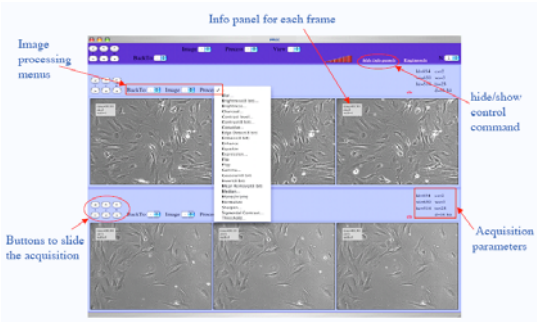
The goal of the life science team in the SCoPE project is twofold and consists of: building a system able to provide a large number of users with general purpose bioinformatics resources, and using them in a number of specific applications. Bioinformatics tools are designed to take advantage of high performance hardware and are easily accessed through a web interface for quick operations, performed by the vast majority of users, but are also available via Unix level access in the form of an integrated problem solving environment. Most commonly used programs have been made available to the users, including software for sequence analysis and alignment, structure prediction and fast sequence search, among others.

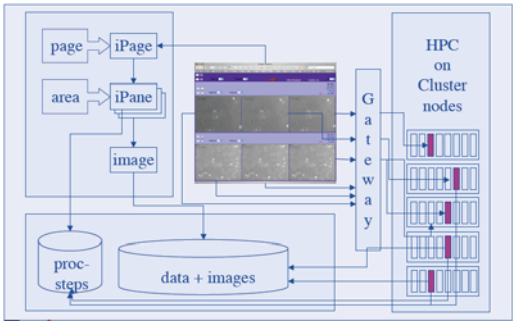
A large number of public databases are mirrored locally, and include most DNA and protein sequence collections, such as EMBL or UNIPROT databases, annotated genome sequences for human as well as mouse, ape, dog, cow and many more eukaryotic and prokaryotic sequences, EST collections and other expression data collections. Databases are accessible through the SRS web interface, but data bank have also been made available to SCoPE users, both in fastA and in BLAST format.

By using these resources, an automatic annotation system has been set up and used in specific computational or experimental projects, to provide analysis of conserved sequences (CST) obtained by comparative genomic analysis. Systematic analysis of human genes allowed to identify and study protein variants as well as gene changes associated with genetically transmitted diseases, such as CFTR. Genome data mining has been used to identify regulatory RNA sequences within completely sequenced bacterial genomes.

Computational analysis of biological images has been used in projects aimed to model the behaviour of cell populations in culture. Large amounts of image data are acquired in 3D observations and in time-lapse studies, and analyzed within the project. A web-based application, IPROC, has been adapted to work in a distributed environment, where image processing is executed in parallel by taking advantage of the available computational resources.

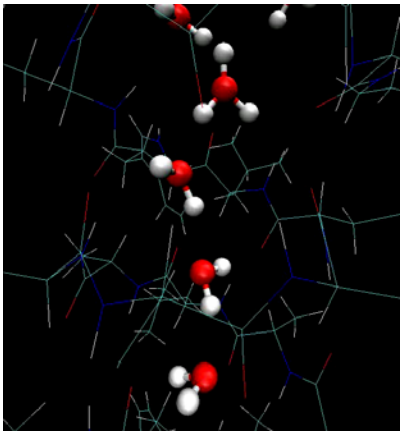
An example of application portal is shown below.





D. The area of computational chemistry [31-45]

Computational modeling of molecular and supra-molecular systems is carried out by the chemistry group in the SCoPE project. Recent developments at the level of theoretical descriptions (e.g. advanced solvent models suitable for dynamical simulations, improved density functionals and basis sets), coupled with efficient algorithms (e.g. linear-scaling implementations) are allowing to achieve good results for small to medium sized structures, and for periodic systems. However, computational descriptions of large, non-periodic structures still represent a challenge. The SCoPE group is focusing on the development and application of integrated computational tools based on hierarchical models, and capable of providing accurate yet computationally feasible descriptions of "difficult" systems such as "soft matter" systems. Crucial elements of the scheme are: a general and powerful electronic model (obtained through the development of original and effective approaches in the framework of the density functional theory); an accurate description of solute-solvent interactions (through mixed discrete-continuum models: see figure below), and the inclusion of the most important effects of nuclear motions (vibrational averaging of physico-chemical observables, reaction rates).



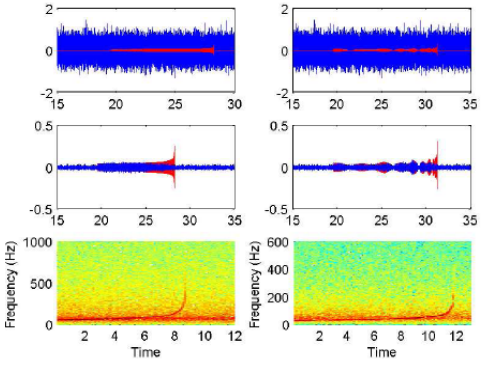
E. The area of gravitational physics [46-48]

Another research area is that of gravitational waves: these are waves due to the presence of a gravitational field, and are predicted by Einstein's General Relativity theory, although nobody has yet observed them. To allow their detection in the frequency interval from 10 to 6000 Hz, Italy and France have built near Pisa, a dedicated observatory, with an instrument, named VIRGO, which is a Michelson interferometer with 3 km arms (see below).

This instrument will be able to detect the effects of gravitational waves coming from supernovae and from coalescing binaries in the Virgo cluster (see simulation below).

The direct measurement of gravitational waves, very small distortion of the space-time, is one

of the most important area of research of modern science.



The figure shows examples of inspiralling gravitational waveforms detection at sampling frequency f =4096 Hz. Top left: classical Neutron Star-Neutron Star binary 1.4 Ms + 1.4 Ms in normalized units with starting frequency 60 Hz, embedded in white noise (SNR=24.5). Middle left: in red theoretical waveform and in blue detected waveform by IIR Ale. Top right: Neutron star+ Black Hole binary 1.4 Ms+10 Ms with maximally spinning BH; Spin=0.99; starting frequency 30 Hz, embedded in white noise (SNR=20). Middle right: theoretical waveform(red) and detected waveform(blue). Bottom right: time – frequency behaviour of the detected waveform by IIR Ale.

Their direct detection of gravitational waves will allow a complete new vision of our universe, including the very first moments of its creation, which are not visible through the investigation of electromagnetic waves. The data which the VIRGO experiment will collect are temporal series at very high frequency, characterized by a very small signal-to-noise ratio; the large number of templates necessary for data analysis using matched- filtering techniques poses problems due to the great computing power needed to perform this task on-line. In fact, as a consequence of the large band of these detectors (some kHz), sampling rates of the order of 20 kHz are foreseen, resulting in a huge amount of data/day to be analyzed on-line (of the order of 10 GB/day). Of course, the analysis of such a large amount of information could be made on-line, but it would be better to select on-line all the data frames which may contain a GW signal. The computational cost depends on the number of parameters considered in the approximation of the phase, on the accuracy of the sampling of the likelihood function (connected with the ability to recover weak signals) and on the actual frequency band to be considered, taking into account the VIRGO sensitivity. Actually, the direct application of the matched-filtering technique to the VIRGO antenna requires a computing power of at least 3 Teraflops starting from a minimum mass of 0.25 M_{Θ} with a SNR recovery of 90% including the computing power necessary for the production of the templates. So these data require computing resources, such as those available in the SCoPE Data Center; but they also require a comparison with similar data from other instruments, such as LIGO in the USA and GEO6000 in Germany. For all these aspect, the Grid model will allow results on a short time frame.

F. The area of astrophysics [49-53]

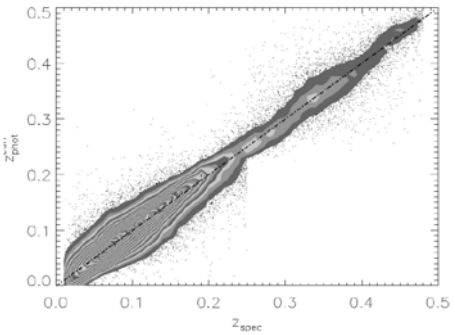
An area of astrophysics research in which GRID infrastructures are essential is that of the so called "Virtual Observatory or VObs". The VObs aims at federating and ensuring the interoperability of all existing data archives (of either observed or simulated data) as well as of data reduction and analysis methods. Since modern instruments and numerical simulations produce data sets in the range of several hundreds of terabytes or petabytes, the VObs poses outstanding computational challenges which find their natural answer in the distributed computing paradigm. At a difference with what happens in other domains, the use of the VObs infrastructure is open to the community as a whole, and therefore poses peculiar security and scalability problems which can be used as a test bed for more general applications.

In this context, the main activity has developed along three research lines: i) the implementation of a procedure based on the use of e-Token robot certificates to allow everyone (even someone who does not possess a personal certificate) to launch specific tasks on the GRID; ii) the implementation of a WEB

application for data mining and exploration in Massive Data Sets (In collaboration with the California Institute of Technology) and, iii) the implementation of several specific scientific problems which effectively show-case the potentialities of the GRID in an astrophysical environment.

The DAME/VOneural platform [49] is a flexible Data Mining and exploration web based framework which allows a generic user to access a large number of machine learning and statistical methods and to run them on the SCOPE-GRID. For its generality the DAME framework is being used also for applications outside of astronomy (such as, bioinformatics and social sciences). The framework interfaces to the distributed computing environment through GRID – Launcher which associates each individual method to a specific robot-certificate.

Among the many science cases produced so far, we wish to emphasize two which have recently been published.



The figure shows the spectroscopic vs photometric redhifts trend for a sample of $3x10^5$ galaxies extracted from the SDSS (rms =0.0186).

Photometric redshifts are one of the most effective ways to estimate the distances of large samples of extragalactic objects. Within the DAME platform it was implemented a specific sub tasks which allows the generic user to derive the phot-z estimate for large samples of objects. The method makes use of an ensemble of machine learning methods trained on a base of knowledge consisting of almost 10^6 spectroscopic redshifts extracted from the Sloan Digital Sky Survey. The resulting catalogue contains photometric redshifts for more than 30 million galaxies derived with unprecedented accuracy (D'Abrusco et al. 2007; Laurino et al. these proceedings).

A second application concerns the physical classification of Active Galactic Nuclei (AGN) using both MLPs and an implementation of Support Vector Machines on the SCoPE Grid. Also in this case the machine learning methods were trained on an extensive spectroscopic base of knowledge and applied to the SDSS photometric catalogues. The results (Cavuoti et al 2009 in preparation) seem capable to open a new era in the exploitation of multiband digital surveys. Preliminary results show in fact that different types of AGN can be disentangled with an accuracy close to 92 %.

An additional result, concerning the simulation of the signatures left by cosmic strings on the Cosmic Microwave Background is presented in Riccio et al. (these proceedings).

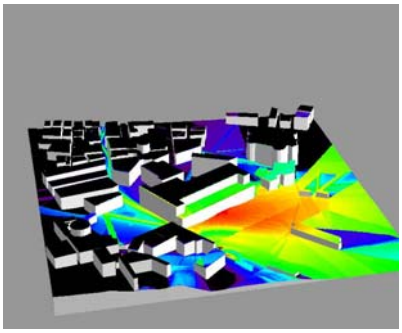
G. The area of electromagnetic waves

Several research activities are being carried out in this area, and one of particular interest concerns the determination of the radio coverage of telecommunication networks (e.g. GSM/UMTS signals) within a city, in order to measure the impact on the environment.

Propagation of electromagnetic waves in the urban environment is in principle a well-posed problem: the electromagnetic field radiated by the sources, for example a single antenna, must fulfill Maxwell equations in the open space, with boundary conditions determined by the buildings walls. But a solution to this apparently simple problem must face the complication of the scattering scenario.

The cities distribution may be considered as a stochastic process, each city being an element of this ensemble. Accordingly, the study of electromagnetic propagation and scattering in the city becomes the problem of searching the

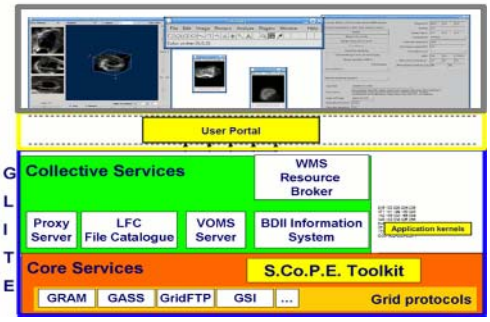
solution of Maxwell equations in a stochastic environment. This task may be pursued on along essentially two lines of thought, either a deterministic or a stochastic one: the former makes reference to an element of the ensemble, the latter exploits the statistical properties of the distribution [54]. In the deterministic approach, an element of the cities ensemble is chosen, namely the particular city of interest. Then, each building is schematized in terms of a parallepipedic structure with plane walls and sharp edges, see figure below, topped with a (usually) flat roof; in addition, average electromagnetic properties of the buildings' walls are either known or postulated.



By accepting this model, ray tracing procedures have been implemented to compute the electromagnetic field everywhere in the open space surrounding the buildings, i.e., in the streets and squares of the city [55-59].
All these studies will allow to plan correctly the implementation and distribution of antennas in the city: the use of the computer grid provided by SCoPE is very convenient for speeding-up the computations and improving the accuracy of the results [60-64] Furthermore, the developed software may be used to interpolate the field values in-between their sampled ones, measured by a limited number of sensors scattered in the city area. In such a way it is possible to evaluate and control the overall exposure level of the population in all points of the city, with a continuous checking of compliance with the Norms, thus reassuring the population in the spirit of the International Health Institute stating that *Health is a state of complete physical, mental, and social well-being and not merely the absence of disease or infirmity.* Concerning this last application, the use of the computer grid provided by SCoPE is fundamental, due to the very large number of electromagnetic sources involved and the convenience of real-time operations.

H. The area of the high level middleware

Special efforts have been devoted to the development of (high level) middleware. It is needed to know, for example, how the new multicore computing platforms affects the efficiency of the underlying numerical kernels [68,69]. Another example of the results obtained is the MedIGrid framework. It is concerned with improvements and enhancements of a medical imaging grid-enabled framework, oriented to the transparent use of resource-intensive applications for managing, processing and visualizing biomedical images. In particular the group has realized an implementation of the MedIGrid PSE, based on the integration of the LCG/gLite and SCOPE-toolkit middlewares, obtaining more efficiency in data management and job submission (see below for software architecture).



The applications allowed by MedIGrid are denoising of 3D echocardiographic images sequence and segmentation of 2D echocardiograph images [65]. Both applications are based on the *Portable, Extensible Toolkit for Scientific Computation* (**PETSc**) library, a suite of data structures and

routines that provide the building blocks for the implementation of large-scale codes concerning the scalable parallel solution of scientific applications modelled by partial differential equations. The group has focused its attention to avoid:

- bottlenecks due to multiple and unuseful Resource Broker (WMS) transits;
- bound to data sizes, imposed by WMS.

Furthermore, the group worked to modify, extend and improve the underlying numerical components to introduce, at the application level, mechanisms to sustain a given level of performance and to survive to a fault during execution flow [66,67].

ACKNOWLEDGMENTS

This work has been funded by the Ministery of University and Research, Italy, under grant no.255/RIC dated 14.02.2006.

The author wishes to thank all the colleagues of the SCoPE project, which have collaborated for the success of the project, as well as to the realization of this paper. Special thanks to the project leader prof. Giuseppe Marrucci.

The realization of the SCoPE infrastructure has been possible thanks to the precious efforts of a number of people, among which Guido Russo, Elena Cesaro (administrative officer), Giovanni Battista Barone (head of the technical staff), Francesco Palmieri (leader of the networking group) and of many young people (among which Silvio Pardi who coordinated the interoperability technical group of the four projects of the PON 200-2006 Avviso 1575/2004) who significantly contributed to the scientific and technical efforts.

The SCoPE group also wishes to thank the staff from DELL, leaded by M. De Leo, who generously devoted most of his time during 2008 to complete the infrastructure in due time.

REFERENCES

[1] The Grid Community at the MIUR web site www.ponricerca.it/Public/grid/F1737/F1737.aspx

[2] The CRESCO e-infrastructure we bsite http://www.cresco.enea.it/

[3] The CyberSar e-infrastructure web site: http://www.cybersar.com/

[4] The PI2S2 e-infrastructure web site: http://www.pi2s2.it

[5] The GRISU web site http://www.grisu-org.it/

[6] The SPACI e-infrastructure website: http://www.spaci.it

[7] Barone G.B., D.Bottalico, V.Boccia, L. Carracciuolo, S.Pardi, M.Scognamiglio, F.Serio, *Middleware applicativo: lo SCOPE Toolkit*, Italian Nat. Conference e-Science2008, 2008.

[8] M.A. Jefri and A.H. Zahed. Elastic and viscous e ects on particle migratrion in plane-poiseuille flow. J. Rheol., 33:691, 1989.

[9] D'Avino G, Hulsen MA, Snijkers F, Vermant J, Greco F, Maffettone PL "Rotation of a sphere in a viscoelastic liquid subjected to shear flow. Part I: Simulation results" J. Rheol. 52, 1331-1346, 2008-

[10] G. D'Avino, G. Cicale, M.A. Hulsen, F. Greco, P.L. Maffettone, "Effects of confinement on the motion of a single sphere in a sheared viscoelastic liquid", J. nonNewt. Fluid Mech., 157, 101-107, 2009.

[11] Carracciuolo L., D. Casaburi, L. D'Amore, G. D'Avino, P.L. Maffettone, A. Murli, *Computational simulations of 3D fluido-dynamic processes in High Performance Computing Environment*, 2009, this volume.

[12] F. Trani, D. Ninno, G. Iadonisi, *Tight-binding formulation of the dielectric response in semiconductor nanocrystals*, Physical Review B 76 , 085326 (2007).

[13] F. Trani, D. Ninno, G. Iadonisi, *Role of local fields in the optical properties of silicon nanocrystals using the tight binding approach*, Physical Review B 75, 33312 (2007).

[14] Yu et al., *Absorption Cross-Section and Related Optical Properties of Colloidal InAs Quantum Dots*, Journal of Physical Chemistry B 109, 7084 (2005).

[15] F. Trani, G, Cantele, D. Ninno, G. Iadonisi, *Tight-binding calculation of the optical absorption cross section of spherical and ellipsoidal silicon nanocrystals*, Phys. Rev. B 72, 75423 (2005).

[16] D. Ninno, F. Trani, G. Cantele, K. J. Hameeuw, G. Iadonisi, E. Degoli, S. Ossicini, *Thomas-Fermi model of electronic screening in semiconductor nanocrystals*, Europhysics Letters 74, 519 (2006).

[17] F. Trani, D. Ninno, G. Cantele, G. Iadonisi, K. Hameeuw, E. Degoli, S. Ossicini, *Screening in semiconductor nanocrystals: ab initio results and Thomas-Fermi theory*, Physical Review B 73, 245430 (2006).

[18] F. Trani, M. Causà, D. Ninno, G. Cantele, and V. Barone, *Density functional study of oxygen vacancies at the SnO_2 surface and subsurface sites*, Physical Review B 77, 245410 (2008).

[19] F. Buonocore, F. Trani, D. Ninno, A. Di Matteo, G. Cantele, G. Iadonisi, *Ab initio calculations of electron affinity and ionization potential of carbon nanotubes*, Nanotechnology 19, 025711 (2008).

[20] P. W. Higgs, *Broken Symmetries, Massless Particles and Gauge Fields*, Phys. Lett. **12**:132, (1964).

[21] P. W. Higgs, *Broken Symmetries and the Masses of Gauge Bosons*, Phys. Rev. Lett. **13**:508 (1964).

[22] P. W. Higgs, *Spontaneous Symmetry Breakdown without Massless Bosons*, Phys. Rev. Lett. **145**:1156 (1966).

[23] O.Brüning et al., *LHC Design Report Volume I: the LHC Main Ring*, CERN-2004-003-V-1, http://cdsweb.cern.ch/record/782076 (2004).

[24] ATLAS Collaboration, *ATLAS: technical proposal for a general-purpose pp experiment at the Large Hadron Collider at CERN*, CERN-LHCC-94-43, http://cdsweb.cern.ch/record/290968 (1994).

[25] L. Perini et al., *Overview of the contributions of the LHC experiments in INFN GRID for bringing the Grid to production quality*, these proceedings (2009).

[26] ATLAS Computing Group, *ATLAS computing: Technical Design Report*, CERN-LHCC-2005-022, http://cdsweb.cern.ch/record/837738 (2005).

[27] G. Carlino et al., *Distributed analysis for the ATLAS Experiment in the SCoPE Project*, these proceedings (2009).

[28] Boccia A, Busiello G, Milanesi L, Paolella G. , *A fast job scheduling system for a wide range of bioinformatic applications*, IEEE Trans Nanobioscience. 2007 Jun;6(2):149-54.

[29] Ausilia Elce, Angelo Boccia, Giuseppe Cardillo, Sonia Giordano, Rossella Tomaiuolo, Giovanni Paolella, Giuseppe Castaldo, *Three novel CFTR polymorphic repeats improve segregation analysis for cystic fibrosis*, Clinical Chemistry, In Press.

[30] Luca Cozzuto, Mauro Petrillo, Giustina Silvestro, Pier Paolo Di Nocera and Giovanni Paolella, *Systematic identification of stem-loop containing sequence families in bacterial genomes*, BMC Genomics. 2008 Jan 17;9:20.

[31] G. Brancato, N. Rega, V. Barone, *A discrete/continuum QM/MM MD study of the triplet state of acetone in aqueous solution*, Chem. Phys. Lett. 453, 202-206 (2008).

[32] V. Barone, M. Biczysko, M. Pavone, *The role of dispersion correction to DFT for modeling weakly bound molecular complexes in the ground and excited electronic states*, Chem. Phys. 346, 247 (2008).

[33] G. Brancato, N. Rega, V. Barone, *A hybrid explicit/implicit solvation method for first-principle molecular dynamics simulations*, J. Chem. Phys. 128, 144501 (2008).

[34] F. Trani, M. Causà, D. Ninno, G. Cantele, V. Barone, *Density functional study of oxygen vacancies at the SnO2 surface and subsurface sites*, Phys. Rev. B 77, 245410 (2008).

[35] M. d'Ischia, O. Crescenzi, A. Pezzella, M. Arzillo, L. Panzella, A. Napolitano, V. Barone, *Structural Effects on the Electronic Absorption Properties of 5,6-Dihydroxyindole Oligomers: The Potential of an Integrated Experimental and DFT Approach to Model Eumelanin Optical Properties*, Photochem. Photobiol. 84, 600–607 (2008).

[36] M. Di Valentin, S. Ceola, G. Agostini, G.M. Giacometti, A. Angerhofer, O. Crescenzi, V. Barone, D. Carbonera, *Pulse ENDOR and density functional theory on the peridinin triplet state involved in the photo-protective mechanism in the peridinin-chlorophyll a-protein from Amphidinium carterae*, Biochim. Biophys. Acta - Bioenergetics 1777, 295-307 (2008).

[37] G. Brancato, N. Rega, V. Barone, *Accurate density functional calculations of near-edge X-ray and optical absorption spectra of liquid water using nonperiodic boundary conditions: The role of self-interaction and long-range effects*, Phys. Rev. Lett. 100, 107401 (2008).

[38] V. Barone, R. Improta, N. Rega, *Quantum mechanical computations and spectroscopy: From small rigid molecules in the gas phase to large flexible molecules in solution*, Acc. Chem. Res. 41, 605-616 (2008).

[39] J. Bloino, M. Biczysko, O. Crescenzi, V. Barone, *Integrated computational approach to vibrationally resolved electronic spectra: Anisole as a test case*, J. Chem. Phys. 128, 244105 (2008).

[40] V. Barone, P. Cimino, O. Crescenzi, M. Pavone, *Ab initio computation of spectroscopic parameters as a tool for the structural elucidation of organic systems*, Theochem 811, 323-335 (2007).

[41] G. Brancato, V. Barone, N. Rega, *Theoretical modeling of spectroscopic properties of molecules in solution: toward an effective dynamical discrete/continuum approach*, Theor. Chem. Acc. 117, 1001-1015 (2007).

[42] A. Pezzella, O. Crescenzi, A. Natangelo, L. Panzella, A. Napolitano, S. Navaratnam, R. Edge, E.J. Land, V. Barone, M. d'Ischia, Chemical, *Pulse Radiolysis and Density Functional Studies of a New, Labile 5,6-Indolequinone and Its Semiquinone*, J. Org. Chem. 72, 1595-1603 (2007).

[43] R. Improta, V. Barone, F. Santoro, *Ab Initio Calculations of Absorption Spectra of Large Molecules in Solution: Coumarin C153*, Angew. Chem. Int. Edit. 46, 405-408 (2007).

[44] M. Pavone, P. Cimino, O. Crescenzi, A. Sillanpaa, V. Barone, *Interplay of Intrinsic, Environmental, and Dynamic Effects in Tuning the EPR Parameters of Nitroxides: Further Insights from an Integrated Computational Approach*, J. Phys. Chem. B 111, 8928-8939 (2007).

[45] I. Borriello, G. Cantele, D. Ninno, G. Iadonisi, M. Cossi, V. Barone, *Ab initio study of electron affinity variation induced by organic molecule adsorption on the silicon (001) surface*, Phys. Rev. B 76, 035430 (2007).

[46] C. Bradaschia, et al., *VIRGO: Proposal for the construction of the very large baseline interferometric antenna for gravitational wave detection*, Proposal to the INFN (Italy) and CNRS (France) (1989, 1992, 1995).

[47] P. Bender, I. Ciufolini, J. Cornelisse, K. Danzmann, W. Folkner, F. Hechler, J. Hough, Y.Jafry, R. Reinhard, D. Robertson, A. Rudiger, M. Sandford, R. Schilling, B. Schutz, R. Stebbins, T. Summer, P. Toubol, S. Vitale, H. Ward, W. Winkler, *LISA: Laser Interferometer Space Antenna for the detection and the observation of gravitational waves*, ESA Pre-Phase A Report, MPQ 208, Max Planck Institut fur Quantenoptik, Munich, Germany (February 1996).

[48] A. Eleuteri, L. Milano, S. Pardi et al., *Adaptive filters for detection of gravitational waves from coalescing binaries*, Phys. Rev. D 73, (2006).

[49] M. Brescia et al., *The DAME/VO-Neural infrastructure: an integrated data mining system support for the science community*, these proceedings (2009).

[50] R. D'Abrusco, et al., *Mining the SDSS data. I. Photometric redshifts for the nearby universe*, ApJ., 663, 752-764 (2007).

[51] O. Laurino et al., *Photometric Redshifts of QSO's*, these proceedings (2009).

[52] S. Cavuoti 2009, Laurea Thesis, University of Napoli.

[53] G. Riccio et al., *Simulations of cosmic string signatures in the CMB*, these proceedings (2009)

[54] G. Franceschetti, R. Marano, F. Palmieri *Wave propagation without wave equation. Toward an urban area model*, IEEE Trans. Antennas Propagat., 1999, 47, 1393-1404 (Vincitore del Premio Schelkunoff per il miglior lavoro su IEEE-AP per l'anno 1999).

[55] M.Franceschetti, D.Riccio, *Wireless Channel Model*, in Wireless Networks: from the Physical Layer to Communication, Computing, Sensing and Control, G.Franceschetti, S.Stornelli, eds., Academic Press, 2006, 161-185.

[56] G.Franceschetti, A.Iodice, D.Riccio, G.Ruello, *A Tool for Planning Electromagnetic Field Levels in Urban Areas*, Proceedings of the Antennas and Propagation Society International Symposium, Monterey, 2004, 2211-2214.

[57] G.Franceschetti, A.Iodice, D.Riccio, G.Ruello, *Pianificazione dei Livelli di Campo Elettromagnetico*, Atti della XIV Riunione Nazionale di Elettromagnetismo, Ancona, 2002, 226-229.

[58] G.Franceschetti, A.Iodice, D.Riccio, G.Ruello, *Determinazione della Radiocopertura per Reti di Telecomunicazioni*, Atti della XV Riunione Nazionale di Elettromagnetismo, Cagliari, 2004, 477-480.

[59] G.Franceschetti, R.Guida, A.Iodice, D.Riccio, G.Ruello, *Verifica di un Software per la Previsione della Radiocopertura*, Atti della XVI Riunione Nazionale di Elettromagnetismo, Genova, 2006, 170-173.

[60] P. Imperatore, F. Moscato, R. Guida, G. Ruello, A. Iodice, D. Riccio, N. Mazzocca, G. Franceschetti, *Strumento su Piattaforma GRID per la Predizione del Campo Elettromagnetico in Aree Urbane*, Conferenza Nazionale Italian e-Science, Napoli, 2008, 117-118.

[61] F. Moscato, N. Mazzocca, P. Imperatore, G. Franceschetti, R. Guida, *COSCA: Una Architettura per l'Analisi di Campi Elettromagnetici su Scala Geografica in Ambienti E-Science Basati su GRID*, Conferenza Nazionale Italian e-Science 2008 - IES08, Napoli, 2008, 148-149.

[62] G. Franceschetti, R. Guida, P. Imperatore, A. Iodice, N. Mazzocca, F. Moscato, D. Riccio, G. Ruello, *Previsione e Controllo della Radiocopertura in Aree Urbane: Un Modello Computazionale Parallelo*, Atti della XVIII Riunione Nazionale di Elettromagnetismo, Lecce, 2008.

[63] G. Franceschetti, P. Imperatore, A. Iodice, D. Riccio, *Radio-Coverage Parallel Computation on Multi-Processor Platforms*, 39th European Microwave Conference, Rome, 2009 (submitted).

[64] G. Franceschetti, P. Imperatore, A.Iodice, D.Riccio, *Computation of the Electromagnetic Field Levels in Urban Environment: a Parallel Model*, to be submitted to IEEE Trans. Antennas Propag.

[65] Carracciuolo L., L. D'Amore, and A. Murli. *Towards a parallel component for imaging in PETSc programming environment: A case study in 3-D echocardiography*. Parallel Computing, 32, 2006, pag. 67-83.

[66] Murli A., V. Boccia, L. Carracciuolo, L. D'Amore, G. Laccetti, and M. Lapegna, *Monitoring and Migration of a PETSc-based Parallel Application for Medical Imaging in a Grid computing PSE*. In P. Gaffney and J. Pool, editors, Grid-Based Problem Solving Environments, Springer Verlag, 2007, pag. 421-432.

[67] Boccia V. , L. Carracciuolo, L. D'Amore, G. Laccetti, M. Lapegna and A. Murli, *Recent advances of MedIGrid PSE in an LCG/gLite environment*, 2009, this volume.

[68] Carracciuolo L., G. Laccetti and M. Lapegna, *Implementation of effective data management policies in distributed and grid computing environments,in Parallel Processing and Applied Mathematics 2007* (R. Wyrzykowski, J. Dongarra, N. Meyer, J. Wasniewski, editors), Lecture Notes in Computer Science 4967, 2007, pag. 902-911.

[69] Laccetti G., M. Lapegna, D. Romano, V. Mele, *Multilevel Algorithms for Multidimensional Integration in High Performance Computing Environment*, 2009, this volume.

Leonardo Merola is Full professor of *General Physics* in the Faculty of Sciences, University of Naples Federico II, and is currently Director of the INFN section in Naples. He is working on the ATLAS project at CERN, and has been involved in many research projects in particle physics and in particular in data analysis

Experimenting with Business Grids

Carmelo Ragusa[1], Francesco Longo[2], Antonio Puliafito[2], Andrea Cicciò[1] and Antonino Isgrò[1]
[1]*COMETA Research Group at University of Messina, 98166, S.Agata, Messina, Italy*
[2]*Department of Mathematics, University of Messina, 98166, S.Agata, Messina, Italy*
{cragusa,flongo,apuliafito,aciccio,aisgro}@unime.it

***Abstract*—Today market demands companies to be agile. Agility is the ability to quickly adapt to the market evolutions at all levels, from business to low level ICT. At application level, the SOA paradigm can address agility requirements through composition and orchestrations of business processes allowing internal and external collaborations, thereby creating new enhanced services. At ICT level, Grid technology is able to deliver such agility too, by providing necessary resources when needed. However, to best support business requirements the combination of SOA and Grid solutions is needed, leading to the creation of a Business Grid. In this paper we presents the results of the S-Sicilia project, a 2-year collaboration between Oracle and the COMETA consortium. We have created a SOA-based business system on top of a gLite infrastructure, combining different services and providing the customers' interface. All requests coming through this module are analysed using QoS mechanisms and addressed to the gLite infrastructure which process them accordingly. Finally, virtualization means have also been integrated to the system to address particular requirements, and offer service customization, higher isolation and security. We have experimented our solution with two different services: the first is a typical enterprise multi-tier web application; the second is a virtualized service for productivity software. Analysis and results have shown advantages and performance enhancements over classical solutions.**

***Index Terms*—gLite, Quality of Service, Service Level Agreement, Business grids, Virtualization.**

I. Introduction

THE market is evolving towards services. Agility is an important capability that each company should have in order to be competitive. At application level, the SOA paradigm has been widely accepted as the technology able to provide service creation, composition and orchestration. The idea to integrate business solutions from different entities, vendors as well as customers, has been a strong aimed desire of many companies. It provides an approach to quickly combine a set of services into a larger distributed application enabling the relative business process with a low implementation cost. Through the SOA, intra and inter relationships are easily creatable. The SOA paradigm is based on different, largely used although not the only ones, enabling technologies such as: Web services, to implement the basic services, and BPEL, to orchestrate the services. Industry sectors such as travel and finance are making large use of the SOA approach, being able to provide services to final customers by just using ready basic services. SOA frameworks are already available on the market and large vendors such as IBM, ORACLE and HP, just to mention some, have developed their solutions. At infrastructure level, agility means the ability to dynamically use resources when needed, without incurring in high initial investments. Cloud computing is the new trend in ICT, where users access transparently their applications from anywhere. Cloud is intended as the non-traditional hosting solution able to provide pay-as-you-go and on demand based easily accessible compute and storage resources. Applications are hosted in data centres able to dynamically quickly scale. Virtualization is the key technology in Cloud computing which allows to separate requests to use physical resources

from the actual physical resources. Using an interface that replaces the physical layer and addresses all the requests allows to present physical resources as multiple logical resources that can be dynamically allocated. In this way, virtualized computing environments can be created, re-sized or moved dynamically. Recently, virtualization reacquired popularity as a solution to reduce cost of ownership and increase resources utilization. Different solutions are available to manage virtual machines such as XEN, VMWare, KVM, VirtualBox, Microsoft Virtual Machine Manager, just to mention some. Moreover, solutions to dynamically connect virtual machines and create virtual clusters over physical resources have appeared on the market. Examples of available solutions are Amazon EC2, IBM Virtualization Manager, Platform VM Orchestrator, Xen Enterprise, OpenNEbula and VM Ware ESX Server. However, although Cloud is extremely flexible it cannot support massive amount of customers, unless of huge investments. On the other hand, one of the tangible achievement of Grid computing is to have created large virtual organizations with a much higher number of physical resources than what a single player could afford. In particular, resources are shared among participants to execute essentially stateless batch jobs. Enabling an existing Grid infrastructure to support the Cloud model can be a winning approach towards the convergence of these technologies for a service-oriented marketplace. Following we present an overview of the system architecture and the two scenarios experimented, web application hosting and virtualization. We also provide conclusions and future directions of our work.

II. Objectives

The S-Sicilia project [1], [2], [3], [4], aims to create real services for companies, that cannot make high investments on physical infrastructure but still want to be active players in the market, as well as for large enterprises, that need extra resources. Our solution is a business SOA-based layer over a gLite infrastructure, offering guaranteed business services.

III. System overview

Due to the always growing dynamicity of the market, estimating resource capacity in advance is almost impossible for companies. There are two principal risks for a company: they under-dimensioning the data-centre and are not able to cope with peak load or they over-dimension it with consequently high investments and most of the time the data-centre is underutilized. Therefore, providing capacity on-demand solutions is a catch for SMEs that do not need to make high initial investments and at the same time have the possibility to be competitive. On the other hand, large enterprises, which usually own large data centre, have the chance to upload their extra load. Such considerations are also valid for the resources that a company uses for its internal administrative or productive activities. Depending on the kind of company there can be short periods of intense activies requiring great amount of resources and long periods of normal activies requiring smaller amount of resources. Moreover, in the case of globally distributed enterprises, it is possible to take advantages of different time zones to share resources between different sites maintaining the dimension of the necessary IT infrastructure within acceptable limits. The S-Sicilia project is a 2-year collaboration between Oracle and the COMETA consortium [5], [6], which aims to investigate mechanisms and techniques to provide Grid-based business services in a commercial context. The COMETA consortium Grid infrastructure is based on gLite. Therefore, providing business services over a scientific middleware is indeed challenging. In the proposed approach we manage services through Service Level Agreement (SLAs), by monitoring service objectives and act accordingly to address the relative requirements. Actions are in terms of service re-configuration, service re-allocation and resources re-allocation. SLAs are also used for accounting and billing operations. At the time when the project started, the most appropriate SLA approach was WSLA, due to its schema structure that well matched the management tasks implemented within the system.

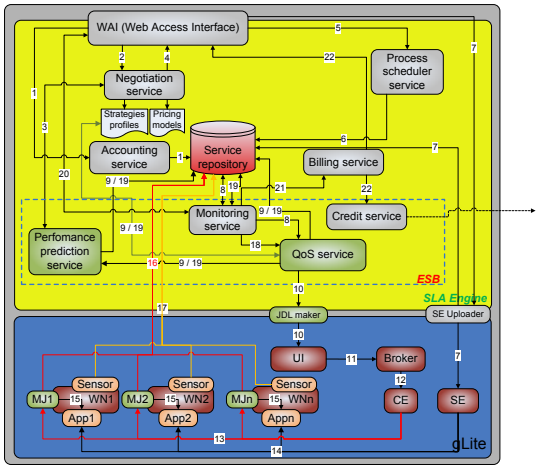


Fig. 1. System architecture.

The XML schema can be modified to suit specific requirements and allows easy creation of SLAs. However, since OGF has recommended WS-Agreement as the way to go for SLAs, one of the future task will be to modify our system to be WS-Agreement compliant. Although our Grid infrastructure is based on gLite, the system was developed to be quite independent from the underlying layer and therefore with little effort it can be adapted to work with other middlewares. Following we present the architecture of our system.

A. Architecture

As mentioned in the introduction, we have combined different technologies to create a system that can offer business services. In Fig. 1 we present the architecture of the system developed.

Two different layers, business and infrastructure, can be easily identified. The business management layer, called SLAengine, was developed using the SOA model. This layer is composed of a set of basic services, orchestrated through BPEL processes. The infrastructure management layer is the gLite middleware, which deals with physical resources. In particular, the interaction between the two layers was developed by implementing two services (JDL maker Service and SE Uploader Service), that integrate the SLAengine with gLite, and a set of Sensors to measure applications resource consumption. The SLAengine manages all SLAs life-cycle, from the definition to the termination, assuring the service compliance by taking any due actions. Customers interact with the system through a Web Access Interface (WAI) exposed by the SLAengine. Whenever a customer requires a service, he/she sends a request using the WAI. The system performs an availability check and depending on results, sends back an offer, in the form of an SLA, to the customer. The customer can accept or reject the offer. Once accepted, the SLA is timely monitored through its life-cycle by the SLAengine. The Service repository component is of central importance for the system. This component is used by the SLAengine services and by the resource sensors. All customers information (credentials, data and SLAs) are stored within this component. The design of the Service repository was made to deal with generic applications. This allows to integrate new applications and therefore new service offers within the system. Services information such as dependencies from other services, addresses, ports, files, etc. are also stored in the Service repository. Moreover, it keeps Sensors information about applications resources consumption. The Service repository maintains also past SLAs archived, which are used to perform statistical calculations for decisions. Finally, another important task of the Service repository is to implement a number of stored procedures performing the logic to manage services, and exposing some functionalities to services that need to acquire stored data. Following is a short description of the other basic services composing the SLAengine:

- Accounting Service: this service registers the user and grants the access to the infrastructure;
- Negotiation Service: this process is started by the user that requests an offer for a service. The Negotiation Service gets the SLA from the user and submits it to the Performance Prediction Service. Based on

the result from the Performance Prediction Service, the strategy and the pricing model in use this service fills the SLA and sends the completed offer back to the user;
- Performance Prediction Service: it returns an estimate of the raw resources that can potentially satisfy the SLA, based on the history of similar SLAs stored within the Service repository;
- Process Scheduler Service: this service receives the user agreed SLA and stores it in the Service repository;
- SE Uploader Service: this service allows the user to upload its application files and stores the files paths, within the SE (the gLite Storage Element), in the Service repository;
- Monitoring Service: this service monitors the SLA life-cycle. It checks if there are SLAs in a new state and sends them to the QoS Service. It also detects if an SLA needs attention and informs the QoS Service. Finally, it determines when an SLA completes and contacts the Billing Service. The operations performed by this service are also implemented by some procedures within the Service repository;
- QoS Service: this service performs a match operation of the user requests with the Grid resources. It also deals with SLAs that might be at risk by taking due actions;
- Billing Service: this service gets the finished SLA, invokes the Credit Service and informs the user that his/her service has been completed;
- Credit Service: this service invokes an external service, usually offered by a bank, which debits the user credit card or account;
- JDL maker Service: this service invokes a script that creates a JDL (i.e. gLite Job Description Language) file for the SLA and sends it to the gLite User Interface (UI);

Sensors are implemented as gLite Jobs and called Monitoring Jobs. They measure resource consumption in terms of computing (CPU and RAM), storage and network, and have been implemented as Linux cron jobs and scripts. They are automatically installed on gLite Worker Nodes (WN) when an application is deployed within the system and their task is to timely read and send resource consumption information associated with the application to the Service repository, so that customer SLA compliance can be monitored. The implementation of the SLAengine was realized using the ORACLE DB 10G and the ORACLE SOA suite. Being SLAs XML based, there was a need to have a native XML DataBase, which has higher performance in managing XML data. ORACLE DB 10g provide such functionality. The ORACLE SOA Suite is a framework that allows creation and management of SOA applications. The system was experimented with two scenarios. A first scenario, regards the hosting of a web-application, dealing with a multi-tier web-application that is deployed on our system and run with QoS guaranties. The second moves towards the Cloud model experimenting with a virtualization scenario enabling customers to use productivity software on a customized and isolated environment.

IV. SCENARIOS

A. *Web application hosting*

Web applications have gained popularity over the past years, due to benefits they bring such as rapid development, scalability, user mobility. Also the possibility to have an initial low investment has given the chance to small businesses to get online and therefore extend their range of actions. However, with low investment only small infrastructure can be built which somehow limits the company chances to grow. On the other hand if a higher investment is considered it may be too risky if the success is not fully accomplished. Giving the companies the possibility to ease their activities from managing their IT-system and potentially lowering their investment will be indeed welcomed. This scenario concerns a hosting solution for a hardware vendor's web application. Customers connect to the vendor's website and make their orders.

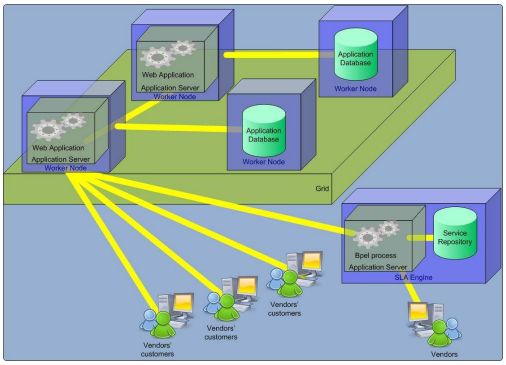


Fig. 2. Web hosting scenario.

Normally, those orders go to the vendor's system, running on an application server, that processes them. We have created an environment to host such application by installing application servers and DBs on the Business Grid nodes (see Fig. 2). In this way all customers' orders are processed by the vendor's system running on the application servers hosted on our Business Grid solution.

This scenario allows experimenting with such type of businesses to show the ability of the proposed system to cope with them. The ORACLE SOA suite comes with a demo application, which is a commercial application for a hardware vendor website implemented through a BPEL process. This demo was chosen for our scenario. Basically, a potential online vendor accesses the infrastructure through a web interface by filling a form with his/her requirements. The system verifies that the service can be performed and sends an offer. After accepting the offer, the vendor sends his/her application in a WAR/EAR format along with the database scripts that are needed to create the necessary DB tables and procedures. Those files are then stored on the gLite SE. The monitoring service detects the new SLA and informs the QoS service. The QoS service chooses the resources (i.e. gLite WNs) to run the service based on the indications of the Performance Prediction Service. The deployment phase then starts by performing an installation of the vendor's application, a DB creation and relative configurations in a fully automated way. After the deployment phase, sensors are attached to the application and its services. Those sensors then send data read to the service repository, which are used by the Monitoring Service to assess the SLA compliance during its lifecycle. If current WNs are not enough to guarantee the SLA requirements a dynamic workload balancing is performed by the Monitoring and QoS Services adding more worker nodes as needed to fulfill the SLA constraints. Viceversa, if an SLA, considering its actual load, has assigned too many WNs an action to release the WNs that are not necessary is performed.

B. Virtualization (Cloud-like)

The aim of this scenario is to start experimenting with the Cloud model by providing a service that allows remote access to customized virtual environments hosting self-contained applications. In particular, the idea is to provide companies the possibility to use external resources for a part of their internal administrative or productive activities. Such activities might involve the use of personal productivity software as well as legacy applications that can become good candidates for such a service. Final users would be the company employees (c-employees) using the application the company manager (c-manager) agreed to install in the customized virtual environments. The guest OS on which we installed the applications was Microsoft Windows Server 2003 while the VirtualBox hypervisor was installed on the gLite WNs to enable the virtualization solution. Fig. 3 gives an overview of this scenario.

The c-manager subscribes the service through the WAI, choosing among three different service classes:

- Gold (one user per VM)
- Silver (three users per VM)
- Bronze (five users per VM)

In order to allow the infrastructure to evaluate the maximum number of virtual machines that can contemporaneously be activated for that

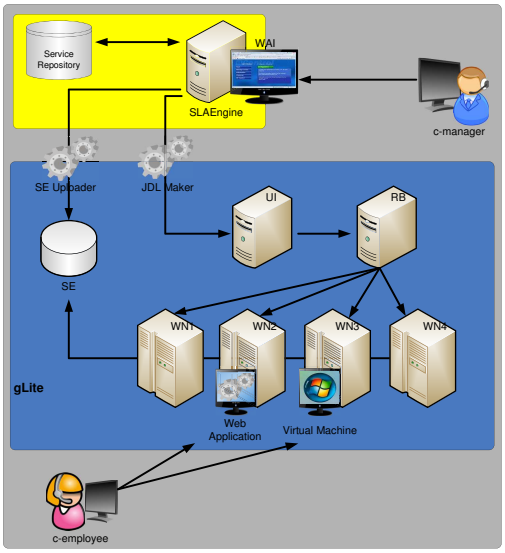


Fig. 3. Virtualization scenario.

particular SLA the c-manager has to indicate also the maximum total number of c-employees that has to be served at a time for the agreed service. We have used Oracle Application Server to host a simple web application that acts as a single access point allowing users that need to access the system to reach the correct VM. The access to the remote VM is possible thanks to VirtualBox that acts like a Remote Desktop Server allowing the connection through any RDP client. Based on the information provided by the Performance Prediction Service, the Negotiation Service sends an offer back to the c-manager. After accepting the offer, the c-manager has to download the image of the basic environment which he/she will customize with the needed applications and with the accounting settings that will allow c-employees to access the service. Following, the c-manager can access a personal control panel through which owned SLAs can be managed and new ones activated. Through the control panel the c-manager has to upload the modified image to the system that stores it in the SE, ready to be used at the moment of the VM activation. The image is tightly coupled with the particular SLA for which it has been modified and the system stores the information about this correlation in the Service repository at the time of the upload to the SE. We are studying the possibility of storing the image in a encrypted way to assure customer data confidentiality. After the image upload phase and at the appropriate time stated in the SLA, the system starts all the mechanisms that activate the c-manager agreed services. Based on the current Grid state, the QoS Service looks for the WNs providing the needed services, that are able to meet the SLA requirements. The Virtualization scenario needs initially two WNs with particular services installed:

- A WN containing the Oracle Application Server, to host the web application through which c-employees are able to access the VMs;
- A WN with VirtualBox, to host the first VM.

After the JDL maker Service is invoked, it launches two grid jobs that land on the identified WNs and activate the web application and the VM. The jar archive of the web application and the VM image for the SLA are downloaded using GridFtp service from the SE to the WN. After the activation of the relative service, a Monitoring Job is launched to monitor the consumed resources and the current number of connections to the VM. The information collected are sent to the Service repository. Further important information is stored in the Service repository, such as the URL of the web application that manages the connections to the VMs and the port number on which the Remote Desktop Server is listening for all the active VMs. At this stage, c-employees can access the VMs and use the application. Thanks to the URL, which is communicated to the c-manager through his/her personal control panel, c-employees can connect to the remote VMs. The web application that manages the connections asks the QoS Service the VM to which the connection has to be redirected and a certified applet is used to launch a Remote Desktop client that automatically connects to

the correct port. The number of current connections to the VMs is periodically compared to the maximum number of contemporary users per virtual machine expected by the SLA and if it is necessary a further WN is sought to launch a new VM. Of course if the total maximum number of c-employees is reached no more VMs are launched. If at any given time all c-employees connected to a particular VM close their connection and the VM is no longer needed, the correspondent WN is freed. The differences between the original VM image and the image after c-employees' connections are stored in the SE and merged to the original image when no c-employees are connected. If changes in the original image are confined to the c-employees' accounts no conflicts are possible. The scenario presented demonstrates how our system is able to deliver Cloud services, transparently to the final users over a well known Grid infrastructure. Capacity is delivered on demand thanks to Monitoring Jobs that timely send needed information and the SLAengine able to take quick actions to assure SLA compliance.

V. Conclusion

This paper presented the S-Sicilia project. Issues related to Business Grids were discussed. Our system aims to deliver solutions targeting SME companies with specific needs and can be a point of reference for the development of the business mechanisms that will lead to the global adoption of the Grid paradigm. However, large enterprises can also benefit when extra resources are needed. We have presented the system, which sits on top of a Grid infrastructure based on the gLite middleware. The Grid infrastructure is mainly used for scientific applications, but we have extended it to provide business solutions. The business layer is based on a SOA approach. Basic services composing our system were introduced. Addition of sensors at gLite level were also introduced. Users access the system through a web interface, establishing a contract containing the service requirements, parties commitments, service performances, terms of service violation and service price. A web application hosting scenario was also shown. In this scenario we offer a service that allows a web vendor to have his/her web application hosted on our infrastructure. Our system presented already a certain level of flexibility, particularly at application level where the SOA paradigm was used for the creation of a business layer. However, more flexibility at infrastructure level was needed to enable full agility. Therefore, the Cloud model was implemented by integrating virtualization services. The virtualization scenario presented aims to offer services for self-contained application such as productivity software and legacy applications. A company can subscribe an SLA to provide its employees with customized VMs, running the required application. Our system is able to provide capacity on demand as final users access and use their environments, in an isolated and secure way. The future steps of our work will be to implement some of the logic behind the scenes. For example, we are working on a mapping mechanism that will allow to get the raw resources based on high level requirements. Tests are being carried out for the purpose. Furthermore, we want to assess this solution over different Grid domains and measure the relative performances. Moreover, we want to expand the range of virtualized service, to offer virtual data-center solutions. This means providing virtual Application Server, virtual clustered DB, virtual balancers, etc. Finally, an important aspect that we are working on is the use of a different approach for the definition of SLAs. Currently we are using WSLA of IBM, but the adoption of WS-Agreement is under developement. WS-Agreement is getting more consensus from the research community. Recent works [7], [8] have shown how it can be extended to allow service runtime re-negotiation.

Acknowledgment

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research

(MIUR) within the Piano Operativo Nazionale Ricerca Scientica, Sviluppo Tecnologico, Alta Formazione (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] "The S-Sicilia project," http://it.wikipedia.org/w/index.php?title=Cubrial&oldid=3406629.

[2] C. Ragusa, S. Arinisi, F. Longo, and A. Puliafito, "A grid-based infrastructure for business applications," in *Proceedings of the Symposium "GRID Open Days at the University of Palermo" Palermo, Italy 6 - 7 December 2007*, Apr 2008, p. 205 211.

[3] C. Ragusa, F. Longo, and A. Puliafito, "On the assessment of the s-sicilia infrastructure: A grid-based business system," in *Grid Economics and Business Models*, Aug 2008, pp. 113–124.

[4] ——, "Experiencing with the cloud over glite," in *To appear on proceedings of ICSE09, 31st International Conference on Software Engineering, Vancouver, Canada, May 16-24, 2009*.

[5] "The Cometa Consortium," http://www.consorzio-cometa.it/.

[6] "Oracle and COMETA Consortium Combine Approach to Grid Computing for Sicilian Pilot Project," www.oracle.com/corporate/press/2006_dec/cometa.html.

[7] G. D. Modica, O. Tomarchio, and L. Vita, "A framework for the management of dynamic slas in composite service scenarios," in *ICSOC Workshops*, 2007, pp. 139–150.

[8] G. Di Modica, V. Regalbuto, O. Tomarchio, and L. Vita, "Enabling re-negotiations of sla by extending the ws-agreement specification," *Services Computing, 2007. SCC 2007. IEEE International Conference on*, pp. 248–251, July 2007.

Dr. C. Ragusa received the Laurea degree in Computer and Electronic Engineering from the University of Catania, Italy, in 2001 and the Ph.D. Degree in Communication Systems from the University of Surrey, U.K., in 2005. He worked as Research Engineer at the IT Innovation Centre, University of Southampton, U.K., on business models for grid computing. He's currently a Researcher of the Cometa Consortium at University of Messina. His research interest are in the areas of distributed computing, Grid, service management, SOA and code mobility.

Mr. F. Longo received his Laurea degree in Computer Engineering from the University of Messina, Italy, in 2007. From settember 2007 to june 2008 he worked in the University of Messina within the PON Project PI2S2. In june 2008 he received a Master's degree in Open Source and Computer Security. He is currently attendig his Ph.D studies in "Advanced Technologies for Information Engineering" at the University of Messina. His interests are in the areas of modeling formalisms, in particular Petri nets, model checking, distributed computing and Grid.

Dr. Antonio Puliafito is a full Professor in Computer Engineering at the University of Messina, Italy, and associate to the Cometa Consortium. He is currently the responsible for all the ICT and e-learning activities for the University of Messina, the director of the RFIDLab, a joint research lab with Oracle and Intel on RFID and wireless, and the director of the Centre on Information Technologies Development and Their Applications (CIA). His interests include parallel and distributed systems, networking, wireless and GRID computing.

Mr. Andrea Cicciò received his Laurea degree in Computer Science from the University of Messina, Italy, in 2006. From November 2007 to October 2008 he worked in the University of Messina within the PON Project PI2S2. In June 2008 he received a Master's degree in Open Source and Computer Security. He is currently working in InQuadro within CC ICT-SUD project regarding RFID Technologies. His interests are in the areas of DataBases, Virtualization, Grid and Web Applications.

Mr. Antonino Isgrò received his Laurea degree in Informatics and Telecomunications Engineering from the University of Messina, Italy, in 2007. From settember 2007 to october 2008 he worked in the University of Messina within the PON Project PI2S2. He is currently attendig his studies in Computer Engineering at the University of Messina and he collaborates in CC-ICT SUD within a project based on RFID-W5 tecnologies. His interests are in the areas of Business Intelligence, Databases and Web Applications.

A Model for XaaS Financial System

Gianni Fenu[1], Francesco Maria Aymerich[1] and Simone Surcis[2]

[1]*University of Cagliari, Italy, fenu@unica.it, aymerich@unica.it*
[2]*COSMOLAB Consortium, Cagliari, Italy, surcis@sc.unica.it*

***Abstract* — Cloud computing is gaining popularity to the extent that the new "XaaS" service category introduced will gradually take the place of many types of computational and storage resources used today. In this regard, grid computing, on which the large scale supply of cloud services is based, will play a key role in defining how those services will be provided.**

This paper concerns the analysis and preliminary design of a real time financial system that relies on cloud computing technologies for performing macroeconomic analysis and forecasts of the financial markets and their instruments.

Cloud and grid paradigms generate different added values. Some of these are examined in detail in the paper, for instance the possibility of controlling load balancing correlated with the number of analyses and forecasts requested, and times, considering the different opening hours of financial markets worlwide.

This work uses the results obtained in the Cybersar Project managed by the COSMOLAB Consortium (Italy). The system analyzed and described herein will be implemented in the Cybersar Computational Grid by autumn 2009.

***Index Terms* — Cloud Computing, Distributed Computing, Financial Data Processing, Grid Computing.**

I. INTRODUCTION

BELIEVED by some to be a newly coined term recently introduced by the media, but actually brought to the foreground by Google's CEO Eric Schmidt, cloud computing is likely to fast become one of the leading technologies in the ICT area. This new computational paradigm, being developed by IBM and Google, will open the doors to Web 3.0.

Cloud computing is about computing over the Internet. The large computing and storage capacity offered by Grid technology, led to another service category being developed. At the time the services were not grouped into a single category, for example web-mail services, web-office automation, web folders for images and so on and only later were they defined as cloud services once it was established they had features in common. Subsequently the cloud computing paradigm defined as a hybrid model for exploiting the resources provided by computer networks.

Cloud computing has both computational and sociological implications. From a computational standpoint cloud computing can be described as a subset of Grid computing using special shared computing resources. Thus it is a hybrid model that exploits computer networks resources, chiefly Internet, improving the features of the client/server scheme. In sociological terms, it changes the way users work, as hardware and software resources are delocalised, thus the user interacts with the "clouds" on-line, instead of in the traditional stand-alone mode. Gartner maintains that cloud computing is one of the 10 strategic technologies for 2008 and many companies will compete in this IT sector.

This results used here have been obtained within the CYBERSAR Project managed by the COSMOLAB Consortium, a project co-funded by the Italian Ministry of University and Research (MUR) within the 2000-2006 National Operational Programme "Scientific Research, Technological Development, Higher

Education" for Italy's Objective 1 regions (Campania, Calabria, Puglia, Basilicata, Sicily, Sardinia) – Board II, Measure II.2 "Information Society", Action 2 "High performance calculation and simulation systems.

One of the aims of the CYBERSAR project is to design and develop distributed supercomputing environments for a variety of disciplinary areas, including economics and financial sciences. A real time financial system, based on cloud computing technologies and paradigms, is being developed to enable analysis and forecasts of the financial markets and their instruments.

The enormous potential and clear advantages offered by Grid and Cloud computing prompted the project idea. Though apparently restricted to a few operators, the financial sector actually involves society as a whole; almost everyone has a bank accounts, for example, Bank account policies depend strongly on financial market trends.

Financial studies have shown that forecasts can be performed using collected data and information as well as specialised graphs (i.e.: Candlestick). These similar and mechanical methods, can be computerized to create financial models that work on the existing or purposely created real time financial systems, as described below.

In most European countries, bar Italy with its poor savings culture, personal investments have become commonplace and have contributed to boosting their economy.

The system under development will be thoroughly tested: large scale treatment of real time financial data using parallel and distributed computing techniques and purposely created financial models for financial market analysis and forecasts. These models could also eventually be used, in the absence of a computational Grid.

The system in fact is designed to work both as an internal Grid service (more specifically as a collection of cloud services) and as a generic web-based application, and can be installed on one or more web server.

A. Conceptual Keys To Cloud Architecture

In a cloud computing architecture hardware and software services are assumed to be essentially stored on the web servers, the "clouds", rather than distributed among the single computers connected in Internet.

The user, provided with a device (PC, PDA, Smartphone or some future device), a browser and an Internet connection, is able to access the right cloud providing the services often tailored to their specific needs and/or data requested.

Another innovative feature is that data can be transferred from one's own PC/device onto a "cloud", enabling the user to access information anywhere.

One characteristic of Cloud computing is the automation of different management tasks, doing away with the need for the human management for process allocation to resources typical of a data centre. This feature is important given the type of data stored in a Cloud platform. Users are able store all their data on the web, thus personal data or sensitive information handling will be more problematic. Cloud service providers will therefore have to guarantee automatic data processing and safe storage.

This and many other aspects of this new distributed computing paradigm are drawn from Grid computing studies and technologies. In fact cloud computing can, to some degree, be thought of as the natural evolution of Grid computing. It was conceived to meet the demands of users who had become accustomed to web 2.0 services and perceived the need to move much of their own data onto the web. Also the technologies and networks that support these new cloud services are now sufficiently mature. From a technical standpoint, cloud computing can be viewed as a subset of Grid computing as they share common technologies and maintain the key

concepts of the new distributed computing paradigm. Each cloud is managed by a computational grid, while a Grid is not able to also implement cloud computing technologies.

Cloud computing is often used interchangeably with SaaS (Software as a Service), but the two should not be confused. Clearly Cloud computing is an extended concept of the new computational paradigm that includes various new categories of services, including SaaS.

Therefore Cloud computing architecture can be said to be related with SaaS, being the most suitable mechanism for delivering SaaS.

B. System Design

A preliminary phase of the study and design of the system consisted in identifying the provider of the real time quotes of financial instruments. This creates technical as well as practical and bureaucratic problems. The most obvious choice would be to retrieve data at source, from the world's main stock exchange data centres (at least USA, Japan, China, UK, Germany, France, Switzerland and Italy in our case) to obtain data in a time (0+k), thus with just one step from source to the final platform. But there are other problems which demonstrate that this hypothesis is not feasible for economic reasons and additional bureaucratic impediments. Moreover, the costs would hardly be sustainable and this option offers very few advantages over the second solution, adopted here. In this case the information is retrieved in a time (0+k+j), i.e. in two steps before arriving at the platform. This is done using a specialized data centre that gathers financial information and data from the world's stock exchanges and resells them. The same order of time (0+k+j) is assigned to stock brokers, while private traders going through banks or other financial institutions to trade on the stock exchange, receive information in time (0+k+j+z) with a high value of z, compared to the times k and j. The times of z depend essentially on the technologies and policies adopted by banks and other financial institutions offering real time quotes and/or direct trading instruments.

Table 1. Comparison of information reception times

Type	Time
Stock Exchanges	**0**
Specialized Data Centres	**0+k**
Banks	**0+k+j** (0+k for the National S. E.)
Stock brokers/traders	**0+k+j**
Private traders/Companies that resort to data centres	**0+k+j**
Private traders/Companies that resort to banks or other credit institutions	**0+k+j+z**

However few data centres collecting and reselling real time data to companies and private traders exist. Substantial effort and much time was devoted to studying and testing the possibilities available, depending on the project's scope. Townsend Analytics, a United States company and leading provider of technology-driven solutions for global capital markets, is for instance one company that could be involved in the project. With over 20 years experience in this sector, Townsend Analytics has made major strides forward, also providing real-time streaming data over the Internet.

Townsend Analytics' flagship product is RealTick, a complete platform for market analysis and direct trading (used by banks in a number of countries).

Townsend Analytics has developed an application programming interface (TAL Toolkit API). This Toolkit enables individuals and institutions to access all those tools required to create custom applications for connecting to Townsend Analytics' real time financial data servers for retrieving price and news data. Townsend Analytics is not the sole

company who supplies APIs. This service is a prerequisite for developing the system as it provides efficient and widely tested communication: Thus the attention can be focussed on the other technical aspects of the system.

During the implementation phase of the system the new Google Finance, which also proposes an API to interface its own data, will also be examined. At present these data are displayed in real time only for the NYSE market and only during opening times. For the rest of the markets they are displayed with a 20 minute time delay.

Next the system's architecture was defined, in its Grid-based version.

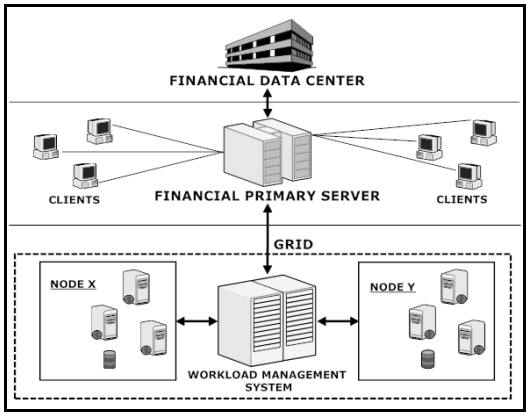


Figure 1. Grid-based system architecture.

The system will comprise an intermediate system that acts as gateway between the financial data center and the main part of the grid managed by the grid middleware. We will call this intermediate system the Financial Primary Server (FPS). GLite middleware is used in grid implementation, but the FPS has been designed to enable interoperability with other grid middleware without too many changes. GLite is the product of the collaborative efforts of more than 80 people in 12 different academic and industrial research centres as part of the EGEE (Enabling Grid for E-sciencE) Project and is currently one of the major grid middlewares in Europe.

Clients are managed by the FPS, so no client communicates directly with the Grid. Almost all analyzed data and processed forecasts will be produced internally to the grid and returned to the clients, through the FPS. The only information flow that the client receives directly, through the FPS and without other "jumps", concerns the real time quotes, so as to respect the time constraint (0+k+j).

Part of the FPS will contain an instruction set enabling jobs to be submitted and more broadly to communicate with the grid. The FPS will communicate with the WMS (Workload Management System), term essentially used in grid middleware (together with "Resource Broker") to indicate the internal Grid system, comprising a group of servers that manage workload submitted to the Grid.

The function of the Grid will be to perform analyses and forecasts over the short-medium term, both of a general nature, thus for all the clients, and specific on demand forecasts on one or more financial quotes. To do this, part of the system's "brain" must be copied inside the grid nodes to inform the grid itself what type of operations to perform for each input. Since it operates using financial models, these copies can be done in one of two ways:

Copy all models into the grid and set up the grid to copy the models into its nodes.

Pass the models as arguments along with the other input data. This is useful when a certain security level is required and it is not desired to enter one's own data into others' computers. This should not create any delays because financial models should not aggravate bandwidth requirements significantly.

After this first set-up phase (which can be automated in both cases without problems) the grid is ready to receive jobs from the FPS.

C. System Design

From an architectural standpoint, the FPS will consist of a series of interconnected computers. For the time being, the single functions and types of devices to be used for

implementation will not be described in detail here but the FPS functionalities/modularities can be summarized as follows:

- Real time Quotation Gateway: this program communicates through the Financial Data Centre API to request/receive data and information. The same program will forward these data to the clients.
- Broker Analyses/Forecasts: this program receives from the Quotation Gateway both real time data, useful for market analysis and forecasts of a general nature, and the analysis/forecast requests from clients for a particular financial instrument.
- Grid Instruction Set: this will be included in the broker functionalities, enabling it to submit analysis/forecast requests to the grid.
- Financial Models Packages: these will be copied into the grid in one of the above described ways.
- Autonomic Computing Element: when the system is not connected to a computational grid, this module enables to perform locally the analysis and forecast operations requested.
- Account Manager: this will manage client authentications.

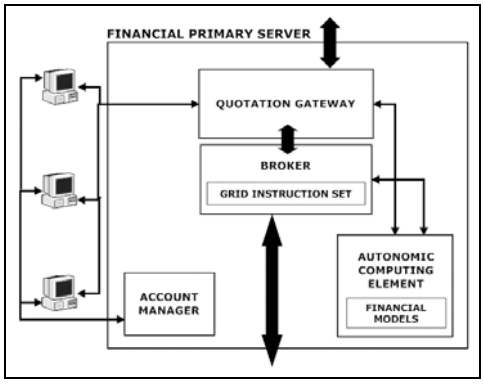


Figure 2. Financial Primary Server Architecture.

D. The Clients

1) Functionality Design

Considering the well-acclaimed research results achieved in grid and cloud computing, it is envisaged that these technologies will be developed for grid-based financial systems, a real possibility in the future. For this reason it is not easy to define a precise target user for the system at the present time. It could be a company LAN, or a financial investors network spread over the country, or web users who want a support tool for on line trading. However in this case, clients will simply be automated "testers" for studying the performance of the system interacting with the grid.

The client software will be web-based and organized into tabs, each tab showing one of the following functionalities:

- Quotes: similarly to a spreadsheet, in each row it will be possible to input and check one real time financial instrument quote. Moreover the tab will show real time quotes for the major market indexes.
- Book: it will be possible to view all the purchases and sales of a selected financial instrument.
- Graphs: it will be possible to display and customize graphs for a selected financial instrument.
- News: it will be possible to display constantly updated news.

Analysis/Forecasts: it will be possible to make a request and to display the result of a selected financial instrument analysis/forecast request or otherwise to view standard forecasts for the reference markets, with the possibility of introducing some analysis parameters.

2) Design decisions resulting from clients analysis

The analysis of clients' behavior plays a fundamental role in designing the functionalities of the entire system, especially

as far as workload is concerned. Considering that some stock exchanges will always be open somewhere in the world, in the 24 hour time span, the system will have to guarantee "always on" operating conditions also at weekends and holidays for short/mid term forecasts.

Every client will have to be able to operate in their own currency. For this reason the system must automatically recognize the country from which the client is operating. Alternatively, every client can choose to operate with a personalized currency. Considered these requirements the system has been designed so that it maintains at runtime all the cross-rates of the world's currencies, updated in push modality similarly to the other financial instruments, in order to avoid consequent losses of time of a conversion executed "on the fly". Another feature of the system is that it estimates workloads based on the hours in the day. This feature was conceived as a result of analyzing daily trading volumes, that exhibit maximums and minimums coinciding with the opening or not of the world's main indexes. Since it is possible to schematize the trading volume trends as a function of hours in the day, it was desired to introduce a second level of load balancing. The latter is additional to the existing level of load balancing carried out from the grid middleware at runtime, and is based on the anticipated reservation of computational and storage resources as a function of predicted workload. In this way, the system will provide additional operational guarantees both to the clients and to the grid that will not risk being over/under utilized. Assuming no geographic restrictions on the use of the system and considering a target of users from all over the world, a possible evaluation of anticipated workload would be that shown in the following diagram with a GMT +6 time zone (USA and Canada):

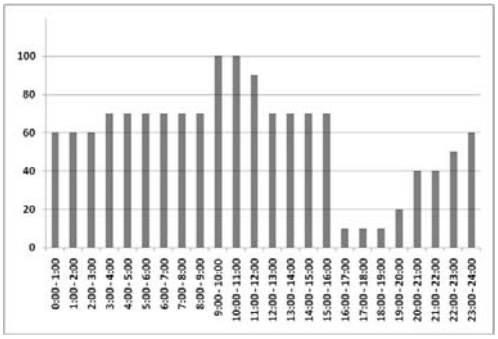


Figure 3. Predicted workload (%) on a daily basis with GMT +6 time zone

E. System Advantages

The system has been designed to operate both in the stand-alone mode and as a computational grid interface. This particular feature allows to take full advantage of parallel computing and to achieve different levels of scalability:

- Scalability of the number of clients: the larger the number of clients, the larger the number of specific analysis and forecast requests. A medium sized computational grid is able to handle hundreds of thousands of requests in acceptable times.
- Scalability of quality of the results: if the system together with the grids, it is possible to use a larger number of financial models over a greater number of CPUs. Since in this case every financial model will be different, even with minor parameter variations, the results obtained should be more accurate and more reliable.
- Scalability of real time forecasts: if the computing is performed inside the grid, the results will be returned more quickly.

Moreover the system is well-adapted for the emerging theories on cloud computing and the described services can be delivered as cloud services. Some well-known financial applications, including the RealTick mentioned above, are disadvantaged by the fact that they

have to be installed in a particular computer (with Windows as OS), undermining their portability. Moreover the operations are executed in a stand-alone mode, that is decreasing the CPU availability of the computer where the software is installed and using the bandwidth just to receive the single real time quotes. Another negative feature is that these applications are ill-suited to mobile devices which instead could use the system services, just connecting to the Internet.

F. Environment Description

The proposed case study has been designed considering the Lambda Grid of the Cosmolab Consortium and its physical-technical characteristics that will be outlined later. We have acquired (in the area and with the aims of the Cybersar project previously mentioned) six computational clusters of varying typologies, each of which comprises tens or hundreds of computational nodes equipped with 64 bit dual-Core processors and with storage systems of various TB. Every computational node has been equipped with a mother-board which supports various boot techniques, of 400 MHz DDR memory RAM and disks that support uninterrupted operation (24 hours a day 7 days a week). As for the software required, every calculation node must work under a standard distribution of linux for x86 or x86_64 with a kernel of 2.6.15 or successive officially released version (vanilla).

The infrastructure is based on complementary computational poles, localized so as to strengthen and to enhance the value of pre-existing research infrastructures and integrated in a Grid architecture. The connections in dedicated fiber optics, allows to experience new computational paradigms, able to combine dynamically distributed resources and to supply a combined peak power of some TeraFLOPS. Each pole is conceived, proportioned and constructed so as to be able to resolve large-scale computational problems. It is specialized in order to meet in an optimal way the specific requirements of important classes of applications.

II. Conclusion

The development of the described system involves a number of stages. At the present stage, the feasibility study has been completed, as well as the high level design (largely described herein) and the detailed design of some significant aspects mentioned in the paper, for example the use of an instruction set to interact with the WMS grid or the possibility to control the load balancing correlated to the time and the day of the week. All those aspects not covered in the paper such as security, authentication etc, are of secondary importance for the project's scope and for this reason are not dealt with here, also considering that they are well-known, standardized and easily usable.

References

[1] Al-Naeem, T., Dabous, F.T., Rabhi, F.A., Benetallah, B., Formulating the architectural design of enterprise applications as a search problem. In *Proceedings of the Australian Software Engineering Conference*, Sydney, 282-291.

[2] Ali, A., McClatchey, R., Anjum, A., Habib, I., Soomro, K., Asif, M., Adil, A., Mohsin, A. From Grid Middleware to a Grid Operating System. In *Proceedings of the International Conference on Grid and Cooperative Computing*, 2006, 9-16.

[3] Aymerich, F.M., Fenu, G., Surcis, S. Programmed Bandwidth & Wavelengths pre-allocation System for Lambda Grids (through lambda grid services). In *Proc. of the Wireless and Optical Communications*, 2007, 111-117.

[4] Barroso, L.A., Dean, J., Holzle, U. Web search for a Planet: the Google Cluster Architecture. *Micro*, IEEE, March-April 2003, 22-28.

[5] Behsaz, B., Jaferian, P., Meybodi, M.R. Comparison of Global Computing with Grid Computing. In *Proceedings of the International Conference on Parallel and Distributed Computing*, Applications and Technologies, December 2006, 531-534.

[6] Choudhary, V. Software as a Service: Implications for Investment in Software Development. In *Proceedings of the International Conference on System Sciences*, January 2007, 209.

[7] Dan, M., The Business Model of Software-As-A-Service. In *Proceedings of the International Conference on Service Computing*, 2007, 701-702.

[8] Dean, J. and Ghemawat, S. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, ACM, January 2008, 107-113.

[9] Ghemavat, S., Gobioff, H., Leung, S. The Google File System. *SOSP*, ACM, December 2003, 29-43.

[10] Grace, P., Hughes, D., Porter, B., Blair, G.S., Coulson, G., Taiani F. Experiences with open overlays: a middleware approach to network heterogeneity, *Operating Systems Review*, ACM, May 2008, 123-136.

[11] Guo-wen, H., Young, W. Chaos and Complexity in the Cumulative Effect of Financial Innovation. In *Proceedings of the International Conference on Management Science and Engineering*, Harbin, 2007, 1635-1640.

[12] Hu, J., Zhong, N. Developing Mining-Grid Centric e-Finance Portal. In *Proceedings of the International Conference on Web Intelligence*, Hong Kong, 2006, 966-969.

[13] Ibrhaim, M. H., Holley, K., Josuttis, N. M., Michelson, B., Thomas, D., John deVadoss. The future of SOA: what worked, what didn't, and where is it going from here?. In *Proceedings of the International Conference on Object Oriented Programming Systems Languages and Applications*, 2007, 1034-1038.

[14] Ivanov, L., Hadimioglu, H., Hoffman, M. A new look at parallel computing in the computer science curriculum. *Journal of Computing Sciences in Colleges*, Consortium for Computing Sciences in Colleges, May 2008, 176-179.

[15] Looker, N., Xu, J. Dependability Assessment of Grid Middleware. In *Proceedings of the International Conference on Dependable Systems and Networks*, 2007, 125-130.

[16] Olsen, E.R. Transitioning to Software as a Service: Realigning Software Engineering Practices with the New Business Model. In *Proceedings of the International Conference on Service Operations and Logistics*, and Informatics, 2006, 266-271.

[17] Raghavan, B., Vishwanath, K., Ramabhadran, S., Yocum, K., Snoeren, A.C. Cloud control with distributed rate limiting, *SIGCOMM Computer Communication Review*, ACM, October 2007, 337-348.

[18] Ramon, N., Ferran, J., David, C., Kevin, H., Jordi, C., Jesus, L., Jordi, T. Monitoring and Analysis Framework for Grid Middleware. In *Proceedings of the International Conference on Parallel, Distributed and Network-Based Processing*, 2007, 129-133.

[19] Ramon, N., Ferran, J., Jordi, T. Should the grid middleware look to self-managing capabilities?. In *Proceedings of the International Symposium on Autonomous Decentralized Systems*, March 2007, 113-122.

[20] Weiss, A. Computing in the Clouds, *networker*, ACM, December 2007, 16-25.

OpenFOAM and Fluent Features in CFD Simulations on CRESCO High Power Computing System

F. Ambrosino[1], A. Funel[1]
[1]*ENEA, Via Vecchio Macello - 80055 Portici (Naples, Italy)*
E-mail: fiorenzo.ambrosino@enea.it, agostino.funel@enea.it

***Abstract*—We present the results of a CFD simulation obtained with Fluent and OpenFOAM, an open source code. Our aim is to compare OpenFOAM capabilities to that of a well known commercial CFD code like Fluent. Both the two codes have been run on CRESCO, the new ENEA high power computing cluster.**

I. Introduction

IN the last years high quality CFD techniques have become necessary and essential tools for tasks such as aircraft industry aerodynamic development, combustion research, fluid-structure interaction problem solving etc. This requires an ongoing struggle for computational efficiency improvements where parallel computing is one of the major issues. As the CRESCO supercomputer has been integrated in ENEA GRID a considerable high power computing can be achieved. Unfortunately, commercial CFD solvers like Fluent have an expensive license cost and generally this cost is directly related to the number of parallel cores needed to run the program. OpenFOAM is an open source CFD solver, it can simulate anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics, electromagnetics and molecular dynamics. Being an open source toolbox it does not need any proprietary license and therefore it is a valid Fluent alternative during all the phases of a CFD solution development.

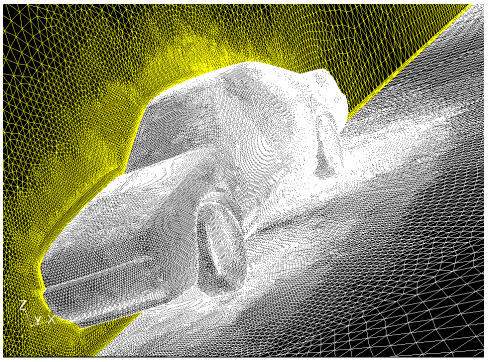


Fig. 1. Fluent mesh snapshot.

II. Problem Specification

THE case study examined here concerns the exterior flow field around a simplified passenger sedan geometry that proceeds with a steady and straight velocity in forward front direction.

The case was produced by ANSYS-Fluent and used as a benckmark of the standard benchmark collection provided by Fluent to test parallel performances on various high power computing systems [1]. It belongs to the "large" class of benchmarks: a viscous-hybrid grid with about 3.6 million prismatic cells is used to adequately model the region of interest. In Fig. 1 is shown the computational domain.

The geometry is outlined in Fig. 2. The car has a velocity v of 27.8 m/s, the pressure p is the atmospheric ambient pressure that is 101325

Pa. The flow is assumed to be steady turbulent. Moreover, due to its low velocity, it can also be considered incompressible.

The equations of motion are:

$$\nabla \cdot \mathbf{U} = 0 \tag{1}$$

$$\mathbf{U} \cdot \nabla \mathbf{U} + \frac{1}{\rho} \nabla p = \nu \nabla^2 \mathbf{U} + \mathbf{g} \tag{2}$$

where $\mathbf{U}$ is the velocity, ρ the density, p the pressure, ν the cinematic viscosity and $\mathbf{g}$ is the gravitational field.

Both for Fluent and OpenFOAM the adopted turbulence model is the standard K-ϵ.

III. OpenFOAM case setting

IN order to run the case with OpenFOAM, the first step is to convert the computational domain in OpenFOAM format. This can be accomplished by using the `fluentMeshToFoam` utility [2]. In Fig. 3 is shown the result for the body car.

The case requires initial and boundary conditions settings for all the involved fields. For applying boundary conditions a boundary is broken up into a set of patches. In OpenFOAM physical boundary conditions are specified by setting appropriate keywords in a file which describes the type of the patches. Moreover, for each involved field a file exists where initial numerical conditions have to be set [2]. In this case the air enters perpendicularly the inlet (y direction) with an initial velocity of 27.8 m/s. The initial pressure field is 1 atm everywhere and at the boundaries we require the condition $\nabla p = 0$. The coefficients used for the turbulence model are $C_\mu = 0.09$, $C_1 = 1.44$, $C_2 = 1.92$, $\alpha_k = 1$, $\alpha_\epsilon = 7.69$. The value of the turbulent kinetic energy is $k = 0.02898$ $\mathrm{m^2 s^{-2}}$ at the inlet, the same value has been set at the outlet for modeling the turbulence in case of backflow. The value of the energy dissipation rate is $\epsilon = 1.8239$ $\mathrm{m^2 s^{-3}}$.

The OpenFOAM solver used is `simpleFoam`, implemented for steady incompressible flow. We adopted as numerical scheme the standard finite volume discretization of Gaussian integration.

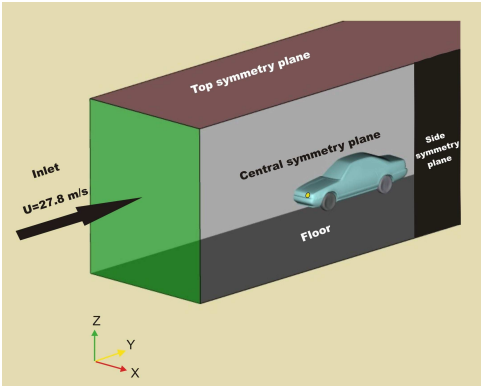


Fig. 2. The geometry.

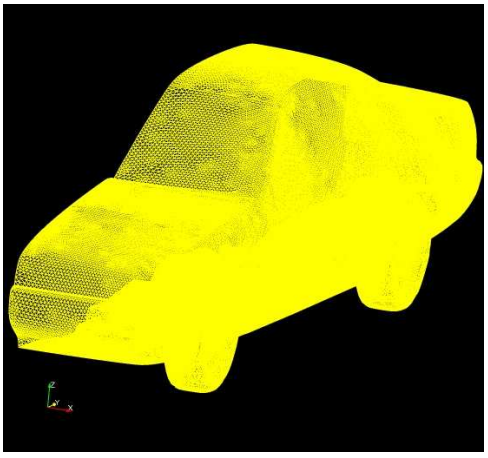


Fig. 3. Body car mesh obtained by converting the original GAMBIT file in OpenFOAM format.

IV. Cross Checking Results

BOTH Fluent and OpenFOAM got convergence. Fig. 4 shows the residuals of the Fluent simulation while Fig. 5 shows the OpenFoam ones.

The viscous-hybrid mesh used had not a good accuracy in modelling the behavior of the flow at the boundary layer. In fact, the y^+ scale value on the sedan surface is of order 100, much more of the viscous sublayer scale value. Fig. 6 shows

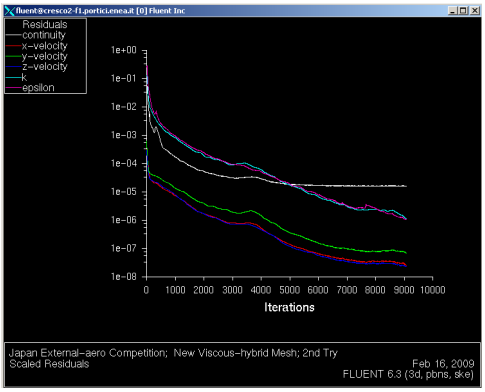


Fig. 4. Fluent residuals.

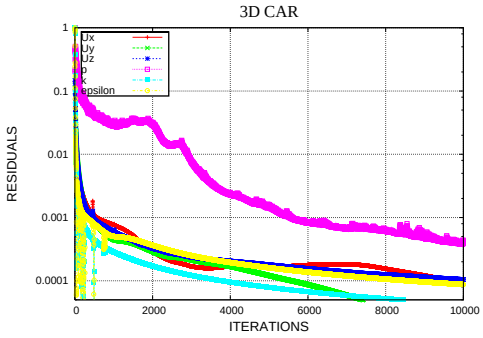


Fig. 5. OpenFOAM residuals.

a map of y^+ on the sedan surface. However, we do not want to study boundary layer effects. Our attention focused on comparing computational performances, thus we considered only global properties of the flow motion.

In order to compare Fluent with OpenFOAM results we shall analyse the shape of the fields on the central symmetry plane. Fig. 7 and Fig. 8 show the contour map of the velocity magnitude field around the sedan for Fluent and OpenFOAM solvers, respectively.

We observe that the magnitude of the velocity fields is of the same order. The velocity field around the car has a similar shape in both the two simulations.

Fig. 9 anf Fig. 10 show the contour map of the pressure field for Fluent and OpenFOAM solvers, respectively. The Fluent picture reports

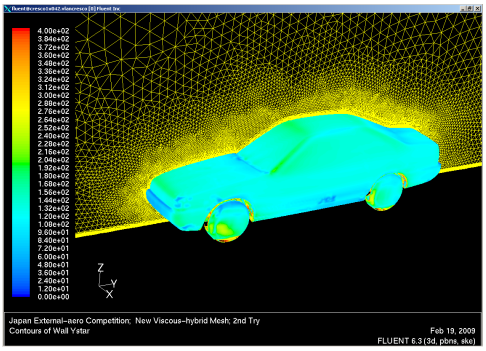


Fig. 6. Map of y^+.

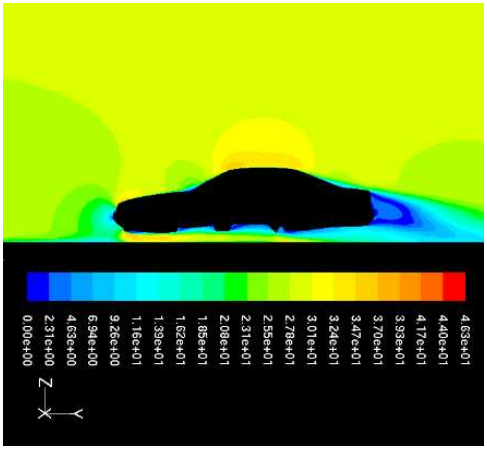


Fig. 7. Contour map of Fluent velocity field.

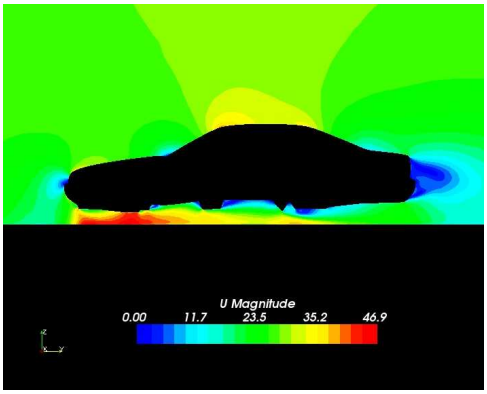


Fig. 8. Contour map of OpenFOAM velocity field.

the pressure values referred to 1 atm, thus negative values indicate a pressure lower than 1 atm. On the other hand, the OpenFOAM picture reports the absolute pressure values as displayed in the scale. In this case an overall agreement is observed, too.

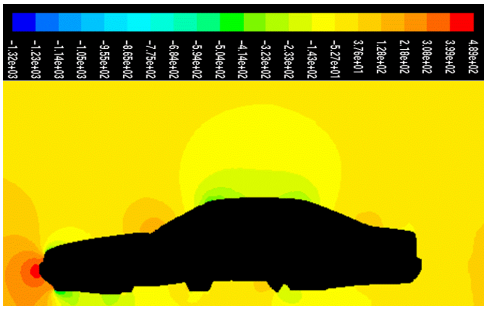


Fig. 9. Contour map of Fluent pressure field.

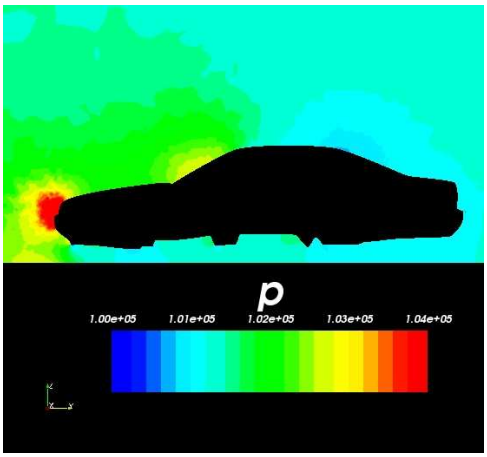


Fig. 10. Contour map of OpenFOAM pressure field.

All simulations have been run on the CRESCO cluster architecture at the ENEA-Portici Research Center. Performances and scalability tests were made both for Fluent and OpenFOAM solvers. Parallel efficiency decreases as the number of computing cores increases. Moreover, parallel efficiency enhances with the mesh size. Fig. 11 shows the speed-up of the simulations up to 512 cores. The speed-up is defined as a ratio of serial to parallel execution time.

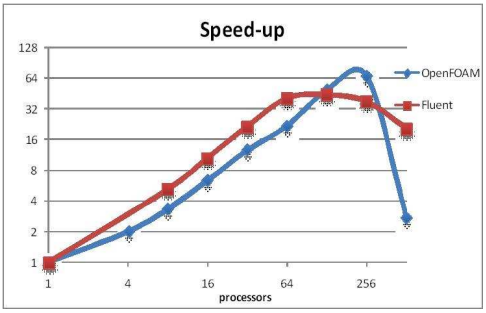


Fig. 11. Speed-up for Fluent and OpenFOAM solvers. The simulations have been run on the ENEA CRESCO high power computing system.

The shape of the speed-up depends on the method used for decomponing the computational domain. In the case of Fluent we used a standard decomposition, as comes with the benchmark, where the mesh is subdivided according to Cartesian directions, while in the OpenFOAM case we used an optimized decomposition method which attempts to minimize the number of common faces between processors. As Fig. 11 shows, the speed-up increases up to 256 cores with the OpenFOAM solver and up about 64 cores with Fluent.

V. Conclusion

In summary we have shown that OpenFOAM may be suitable for running CFD simulations in place of a commercial code like Fluent. The main advantage is that of not having to pay any license cost. This aspect is particularly important when one whishes to run a CFD simulation with many cores. Moreover, being an open source toolbox OpenFOAM is fully extensible and provides the possibility of writing new libraries. We have shown that similar results are obtained by running OpenFOAM with a Fluent benchmark. Tests have been made on the ENEA CRESCO high power computing system which allowed us to establish that OpenFOAM code has a good scalability.

Acknowledgments

We thank Silvio Migliori, Salvatore Podda, Pietro D'Angelo for helping us with their sup-

port. We also whish to thank all the stuff of the ENEA-Portici Research Center.

REFERENCES

[1] http://www.fluent.com/software/fluent/fl5bench/flbench_6.3/problems/fl5l2.htm
[2] http://www.opencfd.co.uk/openfoam/doc/user.html

Numerical Simulation of Turbulent Combustion on CRESCO Platform

E. Giacomazzi[1], F.R. Picchia[2], D. Cecere[3] and N. Arcidiacono[4]
[1]*ENEA, Rome, Italy, eugenio.giacomazzi@enea.it*
[2]*ENEA, Rome, Italy, picchia_r@enea.it*
[3]*ENEA, Rome, Italy, donato.cecere@enea.it*
[4]*ENEA, Rome, Italy, arcidiaconon@enea.it*

***Abstract*—Computational Fluid Dynamics (CFD) is increasing its importance in the design of systems involving reacting turbulent flows like industrial combustion devices. The associated complex physics demands for accurate models and advanced numerical methods. Nowadays supercomputing makes feasible using powerful CFD approaches like LES (Large Eddy Simulations), which is able to meet such demands. CRESCO platform, recently installed in Portici, has absolutely increased computing capabilities of the ENEA research team in combustion. The aim of this work is to show some results of turbulent combustion related problems obtained by means of the ENEA HeaRT code run on CRESCO. HeaRT (Heat Release and Turbulence) is a fully compressible, reactive, unsteady flow solver that implements an original LES subgrid scale model, named Fractal Model. In particular, the main issues of two test cases are discussed. The first one is the Sandia/ETH-Zurich CO/H2/N2 non-premixed unconned turbulent jet ame. This flame is simulated in a three-dimensional axisymmetric formulation. The most important scientific issue related to this flame is its anchoring, analyzed in detail to evidence the controlling physical mechanisms. Besides this, comparison with experimental data is also provided.**

***Index Terms*—CFD, LES, turbulent combustion, parallel computation.**

I. Introduction

THE growing of supercomputers speed in the last few years has produced development and diffusion of numerical models for simulating reactive and nonreactive turbulent ows. In fact, numerics nowadays is the only tool providing multi-scale information at very deep level and taking advantage of powerful mathematical tools (such as those offered by probability theories, chaos and fractals), that are very useful in generating and modeling multi-scale systems. Measurements are still characterized by worse spatial resolution than supplied by numerics. Time resolution is a property of some techniques only, such as LDA (Laser Doppler Anemometry). On the other hand there are some techniques, such as PIV (Particle Image Velocimetry) or the latest stereo PIV, that provide information about scales interaction (but with no time resolution), and could be particularly interesting for understanding and modeling subgrid scale phenomena if coupled to temperature and composition measurements made at the same time of the PIV visualization.

With the aim of benefiting from the numerics multi-scale information, there is much room for turbulence modeling, i.e., for proposing new closures in simulating turbulent ows. Furthermore, due to the great amount of variables and physical phenomena involved, there is even more room for turbulent combustion modeling. These numerical models must be based on the physics and phenomenology of events, and they must be validated. Experimental results, in terms of averaged and uctuating quantities, seems to be the only way to ensure the model worth; to this aim, measurements must be given with error bars, but this is not a common

practise.

RANS/FANS (Reynolds or Favre Averaged Navier Stokes) turbulent closures have been for many years the only tool to numerically simulate complex flows of practical interest. Researchers proposed many models, but not suitable for the simulation of any problem: the crucial point is that RANS/FANS modeling has intrinsic structural limitations due to its modeling of all scales in the same way. Today LES (Large Eddy Simulation) seems to be the answer to this problem: LES is characterized by SGS (Sub-Grid Scale) models that can be more physical and exible closures. At the extreme of numerical simulation there is DNS (Direct Numerical Simulation), but it can be used only for academic ows.

Particularly in the last years LES has been developed and widely used also for practical applications, showing its power to capture the dynamics of a turbulent ow. LES is an attractive tool to simulate turbulent ows; its objective consists in explicitly computing the largest structures of the ow (typically structures larger than the mesh size) and modeling the effects of the small ones. Compared to DNS we lose the description of small scales, whilst, compared to RANS/FANS, we gain the instantaneous resolved elds. Besides, nowadays LES is becoming very useful due to its ability to capture unsteady phenomena. This is particularly true about instabilities: they are due to heat release, hydrodynamic ow eld and acoustic waves, and may induce noise, changes in operational parameters and even destruction of the combustion system. Therefore LES has a strategic role in combustion chamber design and to numerically test passive or active control systems.

In this article the main results of two numerical simulations performed on the CRESCO platform will be presented. The first one regards a non-premixed syngas / air flame present in the SANDIA database. The second one is about a particular combustion regime, named MILD combustion, obtained in a device of Politecnico of Milan burning a mixture of hydrogen and air.

II. Numerical Simulations Set-up

Numerical simulations are performed by means of the in-house code HeaRT (Heat Release and Turbulence) using parallel computers available at the ENEA computational grid [**?**]. The SGS model adopted is the Fractal Model [**?**], [**?**], [**?**]. It is observed that for the first test case a three-dimensional axisymmetric formulation, thus formally two-dimensional, is used; therefore, for that case it is preferable to talk of unsteady simulation rather than LES. In the second case, LES simulation is performed on a sector of 120 degree, due to symmetries in the experimental combustion chamber.

For the present work explicit finite differences (convective and viscous CFL are 0.1 and 0.3, respectively), second order accurate in space (centered) and third order (Runge-Kutta) in time are used to solve the fully compressible Navier-Stokes equations in cylindrical coordinate system. Explicit (non-linear) filtering of field variables is adopted to reduce numerical oscillations due to the centered spatial scheme [**?**], thus avoiding the requirement for a staggered grid. The perfect gas law is assumed as state equation. The computational grids are structured.

III. Simulation of a Syngas Flame

This flame is included in the Sandia/ETH-Zurich database as "Flame A" [**?**], [**?**] and it is a $CO/H_2/N_2$ non-premixed, unconfined, turbulent jet flame. The fuel is injected at $292\,\mathrm{K}$ from a straight circular tubing with squared-off ends (the inner diameter is $4.58\,\mathrm{mm}$, while the outer diameter is $6.34\,\mathrm{mm}$). Its injection bulk velocity is $76\,\mathrm{m\,s^{-1}}$ and the corresponding Reynolds number is 16700. It is observed that the effective maximum velocity at the exit of the injector is about $104\,\mathrm{m\,s^{-1}}$. The air, at $290\,\mathrm{K}$ and wet (molar fraction of water 0.012), coflows at $0.75\,\mathrm{m\,s^{-1}}$. The syngas percent composition is $40/30/30$ in volume. The fuel tubing length is sufficient to make the assumption of fully developed turbulent pipe flow appropriate.

This test case, although its simple geometry, can be useful to test combustion models and

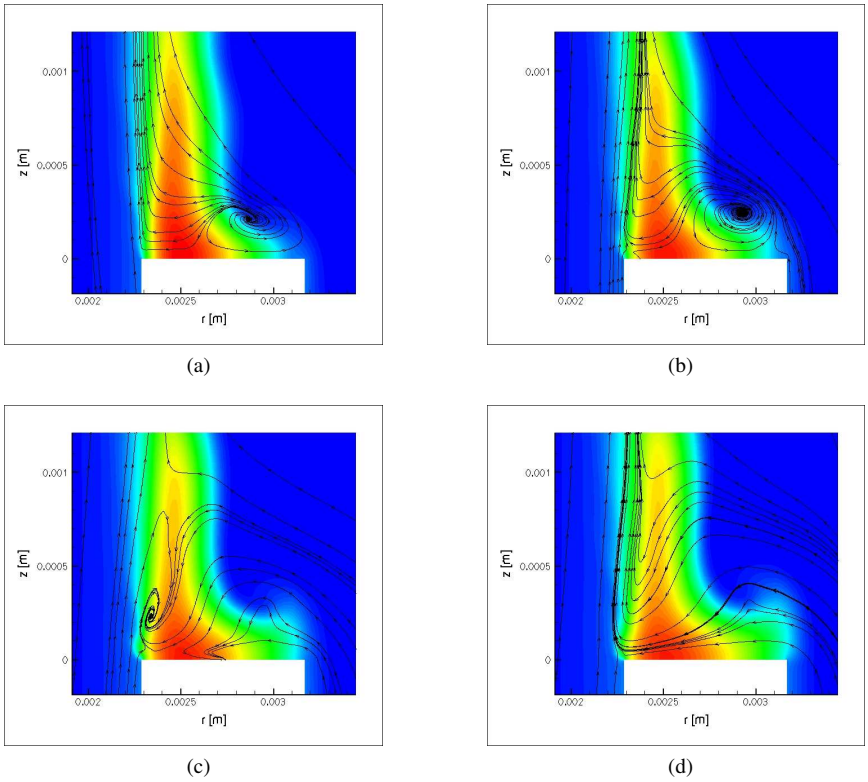


Fig. 1. Temperature contours, details of the shedding and flame anchoring. The delay time between each frame is $\sim 30\,\mu$s.

understand turbulence - chemistry interaction. The use of syngas as fuel facilitates validation because syngas adds only a modest increment in chemical kinetics complexity with respect to hydrogen combustion, while retaining the "simple" structure of hydrogen jet flames.

Experimental observations show that the squared-off nozzle stabilizes the flame thus avoiding a pilot [**?**]. Although not experimentally proved, but shown by the present numerical simulations, the anchoring is likely due to a small recirculation zone produced by the bluff-body. Simultaneous Raman/Rayleigh/LIF measurements of temperature and concentrations, and three-component LDV velocity data are provided by the experimentalists. Measurements for both velocity and scalars (including minor species such as OH) of the same flame are also provided: they are important to understand interactions between fluid dynamics and chemical kinetics. However, the first measurement station is at $45.8\,\mathrm{mm}$ from the injector. Hence, although according to experimentalists the actual flame seems to be continuous and not experiencing any lift off or localized extinction, nothing can be stated precisely about flame dynamics at shorter distance from the inlet.

Although LES offers great advantages with respect to RANS simulations in terms of physical content, particular issues may be hidden by commonly used and accepted "tricks". For example, it is not rare to find simulations of turbulent flames artificially anchored by means of temperature hot spots or flamelet profiles at inflow boundary [**?**]. This practice has the advantage to reduce high spatial resolution requirements in regions of the flow where anchoring is due to very fine turbulent structures that

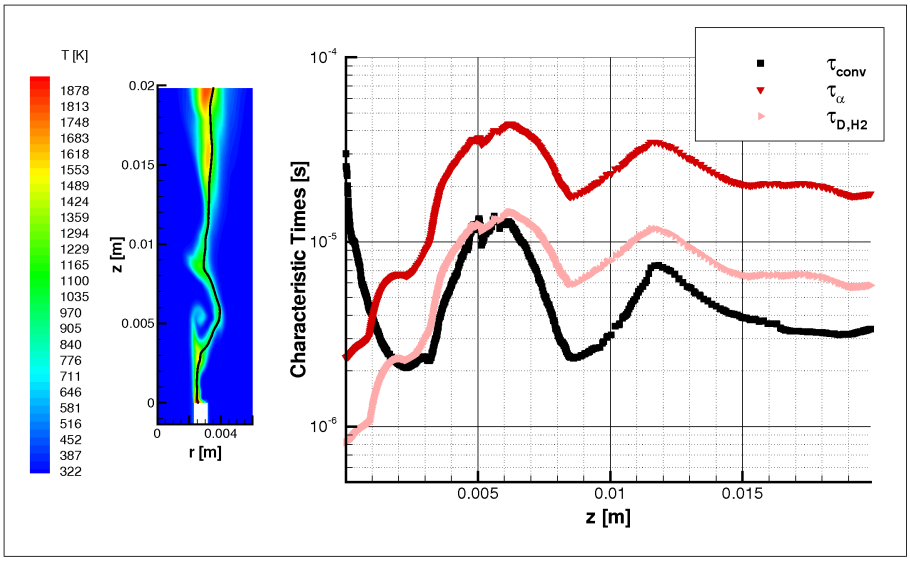


Fig. 2. The competing phenomena in the flame anchoring mechanism as deduced from simulation are fluid dynamic convection (black, lower curve), heat diffusion (red, upper curve) and hydrogen preferential diffusion (pink, middle curve). Their characteristic times are here reported for the instantaneous field shown in Fig. 1a. In particular, these times are sampled along the stoichiometric Bilger's mixture fraction [**?**] iso-line (drawn in black in the flowfield on the left), $F_{BLGR} = 0.295$, and plotted versus the axial coordinate, z.

mix reactants and hot products at so fine scales impossible to be captured on coarse computational grids. On the other hand, this practise inevitably makes impossible to investigate flame anchoring and, at the same time, it changes the dynamic behaviour of the flame, thus reducing the physical value of LES results. Since the flame chosen as test case is well anchored although a very tight bluff-body, it is an interesting case to simulate in order to investigate the main physical mechanisms responsible for its anchoring.

A. *Investigation of Flame Anchoring*

Flame stabilization is the main concern for the numerical simulation of the present flame. In fact, it is likely anchored by means of a small recirculation zone due to the wall of the nozzle tubing, thus avoiding the use of a pilot flame. This experimental convenience may lead to computational inconvenience, in particular when structured grids are used, as in present simulations. The thickness of the tubing is $0.88\,\mathrm{mm}$ and the spatial resolution required to capture the details of the near-nozzle flow is very high. This work shows that these details are important to stabilize the flame.

With the six-step mechanism of Chen [**?**] (based on a skeletal mechanism of 70 reactions with 5 species assumed in partial equilibrium) including main radical species and using a coarse computational grid, the flame blows-off. This means that the fluid dynamic details of flame anchoring in the bluff-body region are not captured at all. This quite unexpected result suggests to analyze the anchoring mechanism for this flame in more detail. Hence, a simulation is performed with increased spatial resolution onto a small region close to the injector with the aim to investigate the physical phenomena that cause a stable and continuous ignition of the reactants as in the experiment.

To understand the actual flame stabilization only a small region, $25 \times 15\,\mathrm{mm}$ (axial and radial sizes, respectively), is simulated with a very fine grid ($10^{-5}\,\mathrm{m}$ spatial resolution close

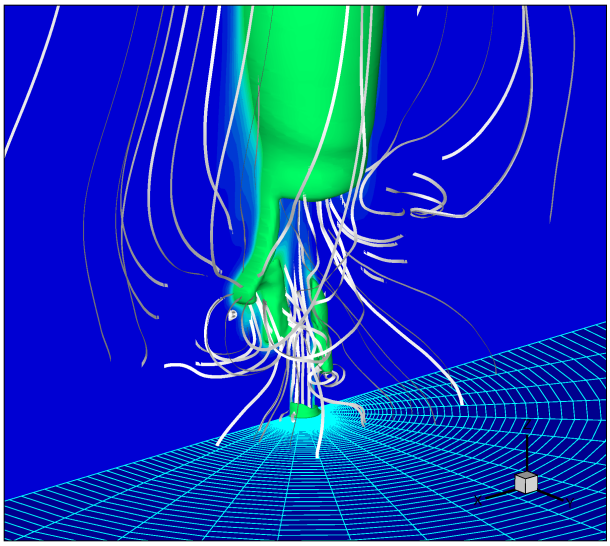


Fig. 3. Three dimensional instantaneous field showing some iso-surfaces of temperature done discontinuous by means of the localized extinctions produced by the stretching of vortices.

to the nozzle). Since Δ, the local characteristic size of the filter, is very small, the ratio Δ/η is small too (~ 3.2, η being the dissipative scale estimated by means of the fuel jet Reynolds number and its diameter assumed as the integral macroscale). It is stressed that this ratio is even smaller in the hot regions due to the increase of viscosity, but is far from DNS resolution in the high speed cold regions (it is observed that the maximum velocity of the fuel jet at its exit is much higher than its bulk velocity, i.e., $\sim 104\,\mathrm{m\,s^{-1}}$ versus $76\,\mathrm{m\,s^{-1}}$). Hence, the SGS eddy viscosity is supposed to vanish or at least to have an order of magnitude comparable to or less than that of molecular viscosity in the low velocity or high temperature regions. This issue was checked and verified: the eddy viscosity is at maximum of the order $10^{-4}\,\mathrm{kg\,m^{-1}\,s^{-1}}$ in the high speed regions of the cold fuel jet, and reaches lower values in the other zones, down to the order $10^{-7}\,\mathrm{kg\,m^{-1}\,s^{-1}}$ in the recirculation immediately downstream of the bluff-body.

Results show a well anchored flame front and seem to confirm what expected by experimentalists, i.e., that the flame is stabilized by a small recirculation zone due to the fuel pipe wall thickness. In fact, as shown in Fig. 1, the end of the injection pipe works as a bluff-body to anchor the flame. In particular, the flame is continuously attached at the inner edge, although the alternate shedding of vortices from the inner and outer edges at $\sim 11\,\mathrm{kHz}$ that causes roll-up of the flame front more downstream. These vortices enhance mixing and anchor the flame. The time-averaged flowfield evidences a recirculation skewed towards the outer edge. This is due to the longer residence time of the outer vortex that brings some coaxial air towards the fuel jet and produce an averaged flame front attached to the inner edge. The strong stability of this "hot spot" close to the injector is also confirmed by the low rms temperature fluctuations in that region. The average presence of the vortex close to the outer edge, as in the configuration of the instant in Fig. 1c is supposed to favour the flame anchoring due to the shorter time scales associated to preferential H_2 mass diffusion and to heat diffusion that enhance mixing and preheating of reactants.

To better understand the actual flame stabilization and discover the physical mechanisms mainly responsible for the anchoring, the char-

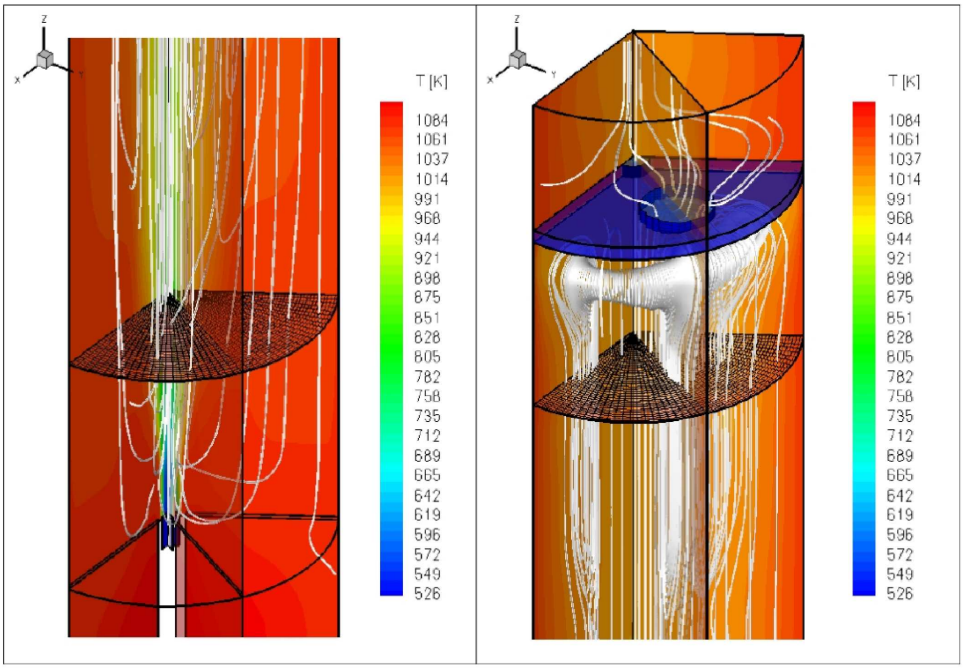


Fig. 4. Temperature contours in the lower and upper regions of the MILD combustor.

acteristic times of physics involved in the simulation are analyzed in the region where the flame is well stabilized. In particular, attention is focused on:

- convective time of fluid motion, $\tau_{conv} = \Delta/U$, U being the local filtered velocity module;
- heat diffusion time, $\tau_\alpha = \Delta^2/\alpha$, α being the local filtered thermal diffusivity;
- preferential H_2 mass diffusion time, $\tau_{D_{H_2}} = \Delta^2/D_{H_2}$, D_{H_2} being the H_2 diffusion coefficient.

It is observed that the turbulent transport properties are used in calculating these times, but they are very close to their molecular value since the eddy viscosity is negligible due to the high spatial resolution.

From the distributions of these times it can be deduced that the competing phenomena in the flame anchoring mechanism are convection of fluid motion, heat diffusion and H_2 preferential diffusion. The competition between these mechanisms is clear looking at axial profiles in Fig. 2 sampled along the stoichiometric Bilger's mixture fraction [**?**]. This figure is associated to the instantaneous field in Fig. 1a. Up to $1\,\mathrm{mm}$ downstream of the bluff-body both heat diffusion and H_2 preferential mass diffusion control the stabilization by enhancing preheating and mixing of reactants. Downstream and up to about $2\,\mathrm{mm}$ the H_2 diffusion is still faster than convection. More downstream convective transport dominates. In the same Fig. 2, it is also possible to note the roll-up produced by the vortex shedding at about $5\,\mathrm{mm}$ from injection.

It is finally observed that the physics becomes more complex when considering three dimensional effects. Three dimensional simulations are currently in progress. Figure 3 is an example of instantaneous field, where three dimensional

vortices produce localized extinctions close to the injector, but the flame is still anchored on average.

IV. Simulation of Hydrogen MILD Combustion

Since the influence of combustion products on the global climate had become a key concern in the discussion about the production of energy, the development of new technologies for a better utilization of fuels in terms of both thermal process efficiency and environmental impact is a major goal of combustion researchers.

The Moderate or Intense Low-oxygen Dilution (MILD) technology offers great advantages in terms of large energy savings with very low pollutant emissions. In this combustion regime the reactants are preheated above the self-ignition temperature and enough inert combustion products are entrained in the reaction region. As a result, the characteristic times of chemical kinetics and turbulent mixing are comparable and the combustion region is no longer identifiable in a flame front but extended over a wide region, so that MILD combustion is often denoted as flameless combustion.

From a historical point of wiew, the technology was first named Excess Enthalpy Combustion, while today it is called High Temperature Air combustion (HiTAC), Flameless Oxidation (FLOX) and MILD. After Prof. Cavaliere [**?**], the MILD combustion regime is achieved when the temperature of the reactant mixture is higher than the mixture self-ignition temperature ($T_{inlet} > T_{si}$) and when the maximum temperature difference with respect to the inlet temperature is lower than the mixture self-ignition temperature in K. The main operation principle for the MILD technique is the concept of hot exhaust gas recirculation, and consequently preheating and dilution of reactants. In fact, while the heat from the exhaust gases causes an increase of the temperature of the reactants, the exhaust gases dilute the mixture, reducing the oxygen concentration and maintaining low temperature in the combustion region. Dilution of reactants is achieved by means of a massive recycle (internal or external to the combustion chamber) of the burnt gases.

Due to its characteristics, a MILD process can easily control and level thermal gradients, thus avoiding the formation of hot spots in the combustion chamber and consequently, increasing the thermal capacity of the system, lowering thermal-NO_x and Soot production without compromising combustion efficiency. Because of small temperature gradients between burnt and unburnt gases, no extinction and ignition regions may be identified in MILD combustion regime. High-velocity jets of fuel, or of combustion air or of both fuel and oxidizer, create a low-pressure region close to the nozzle that promotes the mixing of the unburnt gases with a large amount of hot products. As a result the combustion region is enlarged thanks to a fast dilution of the reactants and the final temperature is well below the adiabatic flame temperature. Since in MILD combustion processes temperature distribution is more homogeneous than classical combustion, materials used for the system are submitted to lower stresses.

A. The Simulated Burner

The experimental apparatus here simulated is located in Politecnico of Milan. The combustion chamber is cylindrical, and the reacting mixture (hydrogen, oxygen, nitrogen, with mass fractions equal to 0.0042, 0.0343, 0.9615, respectively) is injected at $75.83/, m/s$ along its axis through a single high-velocity nozzle with an internal diameter of $3\,\mathrm{mm}$. At the top the chamber is closed with a quartz plate that has three eccentric holes and a central one.

B. Results

The flow topology is characterized by a wide recirculation region which fluctuates in time, as can be observed in Fig. 4. This contributes to preheating and mixing of inlet reactants. Temperature varies only in the pre-heating region of the core jet and shows a flat profile in the recirculation zone. In the experiments, temperature was measured by means of thermocouples

Year	Case	Grid Points	Simulated Time [s]	Δt Time step [s]	N. Iterates	Time per Iterate [s]	Simulation duration [days]	H/W
2004	SYDNEY	1875840	0.03	$7*10^{-9}$	$4.3*10^{6}$	6.5	322	TURAN (32 proc.)
2006	SYNGAS	2256800	0.04	$9*10^{-9}$	$4.4*10^{6}$	3.7	190	LARAN (32 proc.)
2007-2008	MILDH2	2992140	0.03	$3*10^{-9}$	$10*10^{6}$	3.4	394	LARAN (32 proc)
2008	MILDH2	2992140	0.03	$3*10^{-9}$	$10*10^{6}$	0.39	45	CRESCO (512 proc)

Multi-phase Processes

Year	Case	Grid points	Simulated Time [s]	Δt Time Step [s]	N. Iterates	Time per iterate [s]	Simulation duration [days]	H/W
2008	SOMMERFELD	3862528	0.5	$40*10^{-9}$	$13*10^{6}$	0.2	30	CRESCO (1024 proc.)
2008-2009	ISOTHERM	$>5*10^{6}$	0.7	$20*10^{-9}$	$35*10^{6}$	0.15	60	CRESCO (2048 proc)

Fig. 5. Simulation details on different platforms.

at three locations, and these data agree with numerical predictions. Also the NOx predicted at the exit are comparable with measurements (roughly 7 ppm).

Velocity and temperature history were sampled at different locations in the combustion chamber: these data revealed that while high axial velocity fluctuations (20%) are present in the combustion region near the nozzle, the flow experiences a maximum fluctuation in temperature of the order of 5% for a given time windows in a point near the nozzle exit where heat is released. Other points in which high velocity fluctuations are present, were sampled in the combustion chamber and the temperature fluctuations remain lower than the previous value. These maximum fluctuations of temperature are less than those reached in standard flames, and it was expected due to the already high inlet temperature of reactants and to the enthalpy transfer from the hot gas products due to strong recirculation region that promote MILD regime.

V. Performance on CRESCO Platform

Computational fluid dynamics, and especially LES simulation of reactive flows, require huge amount of computational resources, so parallelization is essential.

A Domain Decomposition approach and MPI (Message Passing Interface) are used for the parallel implementation of the HeaRT code. In this approach, the three dimensional computational domain, defined by a multi-block cartesian orthogonal structured grid, is partitioned into a number of sub-domains using a genetic type algorithm to treat whatever geometry. Each sub-domain, that will be handled by one processor, is surrounded by two or more layers of ghost cells to preserve the accuracy required by the solver and to impose physical boundary conditions. The fitness function implemented into the genetic algorithm aims to balance the computational cost between the involved processors and to minimize the amount

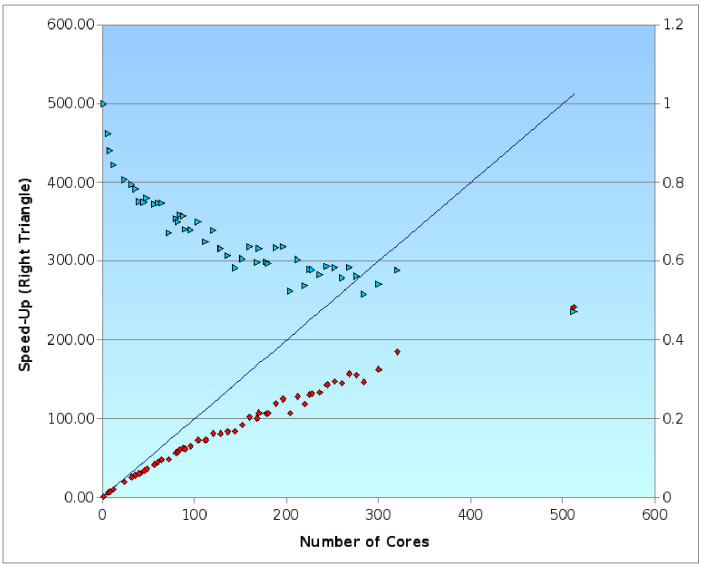


Fig. 6. Speed-up (right triangle) and efficiency (diamond) of the HeaRT code for a specific case on CRESCO platform.

of data to be communicated across the processors boundaries. Because data dependencies are merely nearest-neighbour, at every solvers time step communications occur between a processor and its neighbours to exchange data at their boundaries (through ghost cells). Usually computational time is proportional to volume of domain and communication time is proportional to its surface, but for reactive flow simulation this is not always true. The temporal evolution of a reactive flow can render the initial domain decomposition unbalanced (when parts of the domain are involved in chemical reactions the computational time increases). To avoid code performance's lowering, a dynamic decomposition procedure has been also implemented: this procedure checks the time spent by each processor to solve its own subdomain and, in case of unbalancing, moves the boundaries of each sub-domain until a better performance has been obtained.

The parallel code, written in FORTRAN 95, can be run on different kind of HPC platform (cluster of single or multiple-core machines, SMP machines and so on). HeaRT data concerning different cases on different platforms are reported in Fig. 5 to show how expensive LES simulations of reactive fluxes are. The computational time (expressed by the elapsed time for each time step) was measured over 100 iteration cycles. The data concerning the 3D MILD H2 case (3000000 nodes) shows that HeaRT successfully exploits the computational power of CRESCO platform. The performance data in terms of speed-up and efficiency versus number of cores used, concerning the MILD H2 case running on CRESCO, can be also seen in Fig. 6, Speed-up follows quite well the theoretical value until 64 cores, then the performance decreases but in keeping with the expectations. The efficiency remains around 60% from 200 to 320 cores decreasing to 50% with 512 cores.

VI. Conclusion

Some results obtained on CRESCO platform has been shown. The main conlusion is that this platform contributed to reduce the time to solution, a critical parameter to obtain significant results in an efficient manner aimed to the production point of view of industries. As a matter of fact, this platform is still poor with respect to those available in many international research centres, hence it must grow to maintain high quality levels in CFD.

E. Giacomazzi graduates in Aeronautic Engineering at the University of Rome "La Sapienza" (Dept. of Mechanics and Aeronautics, DMA) in 1996, his Master Thesis being awarded in the "Italian ESSO Award" competition. In 2000 he takes his Ph.D. at DMA in "Theoretical and Applied Mechanics" and his thesys "Modelling and Simulation of Turbulent Combustion" is also awarded in the "Italian ESSO Award". Then, he has several fellowships with DMA and with the Italian Agency for Energy and Environment (ENEA), where he obtaines permanent position in 2006. During these years he participates to several national and international projects in combustion and space propulsion systems; he teaches classes about turbulence and combustion modeling at the AeroSpace Engineering School of Rome; he is tutor for many graduate and Ph.D. students. His expertice covers CFD, parallel computing, experimental diagnostics, control strategies, and theoretical aspects connected with turbulence and combustion.

F.R. Picchia graduates in Physics at the University of Rome "La Sapienza" in 1977. Since 1981 she works at ENEA. She was initially involved in design and development of H/W and S/W systems for acquisition, processing and analysis of data coming from experimental plants aimed to study thermo fluid dynamic phenomena in nuclear reactors components. After she is interested either in advanced software systems applied to different areas (surgery, graphology), either in computing systems management taking up position of responsibility inside the Agency. Since 1997 she is working in parallel computing applied to energy plants. At present she is the leader of a research team, focused on studying combustion processes by means of numerical simulations; more specifically the team is devoted to develop advanced physical and numerical models and to implement them in CFD codes based on Large Eddy Simulation approach.

D. Cecere graduates in Mechanical Engineering at the University of Naples "FEDERICO II" in 2001. In 2002 applies the Photon Correlation Spectroscopy technique for the evaluation of nanoparticle distribution function in laminar diffusion flames. In 2003 he takes his Master in Fluidynamic for Research and Industrial Applications at Sardinian Research Center (CRS4). In 2004 he takes his Ph.D. in Chemical Engineering at the University of Naples "FEDERICO II", disserting his thesis "Large Eddy Simulation of Turbulent Diffusion Flames". In 2007 he obtains a research fellowship at ENEA for studying turbulent combustion with Large Eddy Simulation technique.

N. Arcidiacono works in ENEA since 1983 as an information and software technology technician in scientific computing. Skilled in hardware and software development and computer networks, he worked in team for many different projects aimed to data acquisition systems for energy plant, expert systems for training in surgery, expert systems for traffic management. In the last few years he is focused in parallel programming for distributed and shared memory computer architectures and contributes to the development of CFD software.

Computational Simulations of 3D fluid-dynamic Processes in High Performance Computing Environment

L. Carracciuolo[1], D. Casaburi[2], L. D'Amore[2], G. D'Avino[3], P.L. Maffettone[3] and A. Murli[2]

[1]Consiglio Nazionale delle Ricerche

[2]Dipartimento di Matematica ed Applicazioni
Universitá di Napoli "Federico II"

[3]Dipartimento di Ingegneria Chimica
Universitá di Napoli "Federico II"

***Abstract*—Within the S.Co.P.E. italian national projects, a multidisciplinary collaboration between computational and chemical engineering scientists was established with the aim of developing three dimensional simulations of fluido-dynamic processes. The computing environment relies on PETSc (Portable, Extensible Toolkit for Scientific Computation [1]) components integrated with the TFEM (Toolkit for Finite Element Method [3]) software toolkit, for discretization and solution of the partial differential equation-based model describing the physical processes. We discuss first computational efforts, experiments, numerical and performance results on the basis of selected test cases.**

***Index Terms*—parallel computing, fluid dynamics simulations, PETSc.**

I. Introduction

In many technological applications, solid fillers of different shape, size and chemistry are added to fluids in order to implement functional changes which result in a reduction of process costs or, more frequently, in an improvement of mechanical performance of the final product.

The presence of solid fillers in fluids, which often exhibit themselves a complex rheological behavior, involves other phenomena which, from an applicative point of view have a strong impact on the reliability and the quality of the final product. For instance, the tendency of particles to aggregate and to form ordered structures leads to different filler distribution within the fluid: such behavior should be avoided when the aim of the process is to produce composite materials of high toughness and strength. Therefore, to understand the onset and the evolution of such phenomena is of primary importance for the investigation and construction of materials satisfying the application requirements.

To this aim a multidisciplinary collaboration between computational and chemical engineering scientists in the contest of the S.Co.P.E. Italian national projects was established in order to carry on computational simulations of such 3D fluydo-dynamics processes on high end computing infrastructures. The results of this work can be used as the starting point of future applications in industrial processes such as flow intubation and injection molding.

Scientific simulation is an integrated process. It is also an iterative process, in which computational scientists traverse several times a hierarchy of issues in modeling, discretization, solution, code implementation, visualization and interpretation of results, and comparison to expectations (theories, experiments, or other simulations). They do this to develop a reliable *scientific instrument* for simulation, namely a marriage of a code with a computing and user environment.

One loop over the computational process is related to validation of the model and verification that the simulation process is correctly describing the model. The other loop over the process of simulation is related to algorithm and performance tuning. A simulation that yields high-fidelity results is of little use if it is too expensive to run or if it cannot be scaled up to the resolutions, simulation times, or ensemble sizes required to describe the real-world phenomena of interest. With some oversimplification, algorithm tuning poses the question, *Are we getting enough science per operation?*

During or after the validation of a simulation code, attention must be paid to the performance of the code in the target computing environment. Efforts to control discretization and solution error in a simulation may grossly increase the number of operations required to execute a simulation, far out of proportion to their benefit. Often adaptive strategies must be adopted for the simulation to be execution-worthy on an expensive massively parallel computer, even though these strategies may require highly complex implementation.

It is clear that a scientific simulation depends upon numerous components, all of which have to work.

In this paper we describe first computational efforts towards the development of software tools needed to the computational simulation of fluido-dynamic applications to harness the computational power of high performance computing (HPC) environments.

The computing environment that supports the simulation software relies on PETSc (Portable, Extensible Toolkit for Scientific Computation [1]) components integrated with the TFEM (Toolkit for Finite Element Method [3]) software toolkit, employed to discretize and solve numerical problems deriving from partial differential equation-models.

In particular, we discuss the integration of TFEM modules inside the PETSc parallel data structures, the introduction of the parallelism inside the discretization step, the validation of numerical results and performance gain on the basis of a test case [2].

II. From the Mathematical Model to the Design of the Numerical Algorithm

A. The mathematical model

The problem that we consider consists of a single sphere in a sheared viscoelastic fluid. Assuming incompressibility, negligible inertia, and buoyancy free conditions the governing equations are the continuity (mass balance) and momentum balance equations, plus a constitutive equation depending on the nature of the suspending liquid:

$$\nabla \cdot \boldsymbol{v} = 0 \tag{1}$$

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \tag{2}$$

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + 2\eta_s \boldsymbol{D} + \boldsymbol{\tau} \tag{3}$$

where $\boldsymbol{v}$ is the velocity, $\boldsymbol{\sigma}$ is the stress tensor, p is the pressure, $\boldsymbol{I}$ the unity tensor, $\boldsymbol{D} = (\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T)/2$ is the rate-of-deformation tensor, η_s is the solvent viscosity and $\boldsymbol{\tau}$ is the constitutive extra stress.

The polymer stress, $\boldsymbol{\tau}$, is given written as:

$$\boldsymbol{\tau} = \frac{\eta_p}{\lambda}(\boldsymbol{c} - \boldsymbol{I}) \tag{4}$$

where $\boldsymbol{c}$ is the 'conformation tensor', η_p is the polymer viscosity, and λ is the relaxation time.

We will model the viscoelastic fluid with the Giesekus constitutive equation (for $\boldsymbol{c}$):

$$\lambda \overset{\nabla}{\boldsymbol{c}} + \boldsymbol{c} - \boldsymbol{I} + \alpha(\boldsymbol{c} - \boldsymbol{I})^2 = \mathbf{0} \tag{5}$$

where α is the so-called mobility parameter that modulates the shear thinning behavior. The

symbol ($^{\nabla}$) denotes the upper-convected time derivative, defined as:

$$\overset{\nabla}{\boldsymbol{c}} \equiv \frac{\partial \boldsymbol{c}}{\partial t} + \boldsymbol{v}\cdot\nabla\boldsymbol{c} - (\nabla\boldsymbol{v})^T\cdot\boldsymbol{c} - \boldsymbol{c}\cdot\nabla\boldsymbol{v} \quad (6)$$

A sphere of radius R is located between two plates moving along the x-direction in different sense with an imposed shear rate $\dot{\gamma}$. Shear flow conditions are imposed on the external fluid boundaries.

The sphere is located at the center of the gap and symmetry implies that it can only rotate. We assume that the sphere is torque-free, or "freely rotating", i.e. its rotation is only due to the motion of the surrounding fluid. Thus, the torque-free boundary condition at $r = R$ is:

$$\int_{\partial S_p} \boldsymbol{r}\times\boldsymbol{\sigma}\cdot\boldsymbol{n}\,dA = \boldsymbol{0} \quad (7)$$

where ∂S_p is the sphere surface, $\boldsymbol{n}$ is the normal at the sphere surface, $\boldsymbol{r}$ the position vector from the sphere center and the integral of the local torque $\boldsymbol{r}\times\boldsymbol{\sigma}\cdot\boldsymbol{n}\,dA$ spans the sphere surface. Due to the symmetry of the imposed shearing flow, only the vorticity component z of Eq. (7) is relevant, the other two components being identically zero. In fact, the velocity at the sphere surface is known, and can be written as:

$$\boldsymbol{v}(R,t) = \omega(t)\boldsymbol{e}_z \times \boldsymbol{R} \quad (8)$$

$\boldsymbol{e}_z$ being the unit vector along z. The particle angular velocity, $\omega(t)$, is, however, unknown and it depends on the nature of the fluid.

B. Weak form

The governing equations are solved by the finite element method on a cell containing a single sphere at the center of the cell. A fictitious domain is used [4] together a rigid-ring description of the particle [5], since inertia is neglected. A log-conformation representation for the conformation tensor has been used: the original equation for the conformation tensor $\boldsymbol{c}$, Eq. (5), is transformed to an equivalent equation for $\boldsymbol{s} = \log(\boldsymbol{c})$, leading to a substantial improvement of stability (see [7] for details). To further improve the numerical stability a DEVSS-G formulation is implemented [8] for the momentum balance and the SUPG technique [6] is used for the constitutive relation.

Under these assumptions, a weak form can be stated as follows: For $t > 0$, find $\boldsymbol{v} \in U, p \in P, \boldsymbol{s} \in S, \boldsymbol{G} \in H, \boldsymbol{\omega} \in \Re, \boldsymbol{\lambda} \in L^2(\partial S_p)$ such that:

$$\int_V 2\eta_s \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v})\,dV - \int_V \nabla\cdot\boldsymbol{u}\,p\,dV + \int_V \eta_p(\nabla\boldsymbol{u})^T : \nabla\boldsymbol{v}\,dV - \int_V \eta_p(\nabla\boldsymbol{u})^T : \boldsymbol{G}^T\,dV + \langle \boldsymbol{u} - (\boldsymbol{\chi}\times\boldsymbol{r}), \boldsymbol{\lambda}\rangle_{\partial S_p} = -\int_V \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{\tau}\,dV, \quad (9)$$

$$\int_V q\,\nabla\cdot\boldsymbol{v}\,dV = 0, \quad (10)$$

$$\int_V \boldsymbol{H} : \boldsymbol{G}\,dV - \int_V \boldsymbol{H} : (\nabla\boldsymbol{v})^T\,dV = \boldsymbol{0}, \quad (11)$$

$$\int_V (\boldsymbol{S} + \tau\boldsymbol{v}\cdot\nabla\boldsymbol{S}) : \left(\frac{\partial \boldsymbol{s}}{\partial t} + \boldsymbol{v}\cdot\nabla\boldsymbol{s} - \boldsymbol{g}(\boldsymbol{G},\boldsymbol{s})\right) dV = \boldsymbol{0}, \quad (12)$$

$$\langle \boldsymbol{\mu}, \boldsymbol{v} - (\boldsymbol{\omega}\times\boldsymbol{r})\rangle_{\partial S_p} = \boldsymbol{0}, \quad (13)$$

$$\boldsymbol{s} = \boldsymbol{s}_0 \quad \text{at } t = 0, \quad \text{in } V, \quad (14)$$

for all $\boldsymbol{u} \in U$, $q \in P$, $\boldsymbol{S} \in S$, $\boldsymbol{H} \in H$, $\boldsymbol{\chi} \in \Re$ and $\boldsymbol{\mu} \in L^2(\partial S_p)$, where U, P, S, G are suitable functional spaces. The τ parameter in Eq. (12) is given by $\tau = \beta h/2U_e$, where β is a dimensionless constant, h is a typical size of the element and U_e is a characteristic velocity for the element. Finally, we take the initial value of $\boldsymbol{s}_0 = \boldsymbol{0}$, corresponding to zero initial stress.

Notice that the angular velocity, $\boldsymbol{\omega}$, is treated as an additional unknown and is included in the weak form of momentum equation. Only the z-component of $\boldsymbol{\omega}$ is set different to zero since, for the symmetry of the problem, the sphere can rotate around the vorticity axis only. The torque-free condition is imposed through the Lagrange multipliers, $\boldsymbol{\lambda}$.

C. Numerical model and algorithm

For the discretization of the weak form, we use hexahedron elements and a continuous bi-quadratic interpolation (Q_2) for the velocity $\boldsymbol{v}$, bilinear continuous interpolation (Q_1) for the pressure p, linear continuous interpolation (Q_1) for velocity gradient $\boldsymbol{G}$ and bilinear continuous interpolation (Q_1) for the log-conformation tensor $\boldsymbol{s}$. We discretize the boundary of the particle using the weak constraints. Then, discretization of Eq. (13) gives to:

$$\langle \boldsymbol{\mu}, \boldsymbol{v} - (\boldsymbol{\omega} \times \boldsymbol{r}) \rangle_{\partial S_p} = \int_{\partial S_p} \boldsymbol{\mu} \cdot [\boldsymbol{v} - (\boldsymbol{\omega} \times \boldsymbol{r})] \, dA \quad (15)$$

Regarding the time discretization, initially, the viscoelastic polymer stress is set to zero in the whole domain. Then we solve the equations (9)-(11) and the constraint equation (13) in order to get the distribution of the fluid velocity and the angular velocity of the particle, at the initial time step. At every time step, we perform the following procedure:

Step 1. The log-conformation tensor at the next time step, $\boldsymbol{s}^{n+1}$, is evaluated by integrating the constitutive equation (12). A semi-implicit first-order Euler scheme is used:

$$\frac{\boldsymbol{s}^{n+1}}{\triangle t} + \boldsymbol{v}^n \cdot \nabla \boldsymbol{s}^{n+1} = \frac{\boldsymbol{s}^n}{\triangle t} + \boldsymbol{g}(\boldsymbol{G}^n, \boldsymbol{s}^n) \quad (16)$$

where $\boldsymbol{g}$ is the term which appears in the evolution equation of $\boldsymbol{s}$.

Step 2. The remaining unknowns $(\boldsymbol{v}, p, \boldsymbol{G}, \boldsymbol{\omega})^{n+1}$ as well as the Lagrange multipliers ($\boldsymbol{\lambda}$) can be found by solving the Eqs. (9)-(11) and (13) using the particle configuration and the polymer stress evaluated in the previous two steps:

$$\int_\Omega 2\eta_s \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v}^{n+1}) \, dA - \int_\Omega \nabla \cdot \boldsymbol{u} \, p^{n+1} \, dA + \int_\Omega \eta_p (\nabla \boldsymbol{u})^T : \nabla \boldsymbol{v}^{n+1} \, dA - \int_\Omega \eta_p (\nabla \boldsymbol{u})^T : (\boldsymbol{G}^{n+1})^T \, dA + \langle \boldsymbol{u} - (\boldsymbol{\chi} \times \boldsymbol{r}), \boldsymbol{\lambda}^{n+1} \rangle_{\partial S_p} = -\int_\Omega \boldsymbol{D}(\boldsymbol{u}) : \boldsymbol{\tau}^{n+1} \, dA, \quad (17)$$

$$\int_\Omega q \nabla \cdot \boldsymbol{v}^{n+1} \, dA = 0, \quad (18)$$

$$\int_\Omega \boldsymbol{H} : \boldsymbol{G}^{n+1} \, dA - \int_\Omega \boldsymbol{H} : (\nabla \boldsymbol{v}^{n+1})^T \, dA = 0, \quad (19)$$

$$\langle \boldsymbol{\mu}, \boldsymbol{v}^{n+1} - (\boldsymbol{\omega}^{n+1} \times \boldsymbol{r}) \rangle_{\partial S_p} = 0 \quad (20)$$

The Eq. (16) leads to an unsymmetric linear system whereas in Step 2 a sparse linear symmetric system needs to be solved.

Algorithm 1 shows how **Step 1** and **Step 2** organize themself to discretize and solve the problem: in particular **Step 1** is performed as described at points 6 and 7 while **Step 2** is performed as described at 8 and 9.

Algorithm 1 Algorithm outline

1: Define problem
2: Compute A_0 and b_0
3: $x_0 \leftarrow A_0^{-1} b_0$
4: $x_0^c \leftarrow$ Some initial condition
5: **for** $t \leftarrow 1, num_time_steps$ **do**
6: Compute A_t^c and b_t^c from x_{t-1} and x_{t-1}^c
7: $x_t^c \leftarrow (A_t^c)^{-1} b_t^c$
8: Compute A_t and b_t from x_t^c
9: $x_t \leftarrow A_t^{-1} b_t$
10: **end for**

III. The computational environment and the parallelization details

THE target computing environment we consider to implement Algorithm 1 is a parallel and distributed memory model where we hide, within the PETSc objects, the details of internode communications.

PETSc is a suite of data structures and routines for the scalable parallel solution of scientific applications modeled by partial differential equations. PETSc is flexible: its modules, that can be used in application codes written in Fortran, C and C++, are developed by means of object-oriented programming techniques [1].

The PETSc hierarchical structure, shown in figure 1 [1], allows the choice of problem depending levels of abstraction; it relies on standard basic computational (BLAS, LAPACK) and communication (MPI) kernels, and provides mechanisms needed within parallel application codes, such as parallel matrix and vector assembly routines that allow the overlap of communication and computation, as well as tools for the parallel management of discretization grids. Looking at figure 1 from bottom-up, i.e. towards a growing level of abstraction, Krylov subspace methods and scalable parallel preconditioners are located, as the *"floor"* upon which linear and non-linear systems solvers rely.

The suite of solvers and preconditioners can be enriched interfacing PETSc with several packages, such as, *BlockSolve95*, *SuperLU*, *SuperLU_Dist*, *Hypre*, *MUMPS*.

We integrate TFEM (Toolkit for Finite Element Method [3]) within the PETSc computing environment TFEM. TFEM is the sequential software already employed for the discretization and the solution of this problem. TFEM is modular and its modules, that can be used in application codes written in Fortran90/95, are developed by means of object-oriented programming techniques. It consists of:

- a kernel, which includes object type (i.e. `mesh`, `problem`, `sysmatrix` and `sysvector` objects) definition/construction and tool for Gaussian integration on elements of different shape;
- some *"add-on"*s including tools performing: data visualization, the interface to external linear equation system solvers (PARDISO, MUMPS, etc.), the interface to external software for graphs reordering (METIS), the definition of details related to the discretization of some models (Stokes model, Viscoelastic model, etc.).

Implementation of Algorithm 1 on the target HPC environment means the use of TFEM at steps 1, 2, 6, 8 and of PETSc at steps 3, 7, 9. Schematically, we design the following algorithm:

Algorithm 2 Parallel Algorithm using TFEM+PETSc

1: TFEM: Define problem
2: TFEM: Compute A_0 and b_0
3: PETSc: $x_0 \leftarrow A_0^{-1} b_0$
4: $x_0^c \leftarrow$ Some initial condition
5: **for** $t \leftarrow 1, num_time_steps$ **do**
6: TFEM: Compute A_t^c and b_t^c from x_{t-1} and x_{t-1}^c
7: PETSc: $x_t^c \leftarrow \left(A_t^c\right)^{-1} b_t^c$
8: TFEM: Compute A_t and b_t from x_t^c
9: PETSc: $x_t \leftarrow A_t^{-1} b_t$
10: **end for**

To this aim, it is required:

1) to modify TFEM modules to replace original data structure used in `sysmatrix` and `sysvector` objects by the PETSc parallel objects
2) to introduce the parallelism during the definition of such objects using *"row-block"* distribution (points 2, 6, 8 of the Algorithm 1),
3) to identify the PETSc computational solvers to use (points 3, 7, 9 of the Algorithm 1).

IV. Preliminary results and Conclusions

[1] http://www.mcs.anl.gov/petsc/petsc-as/features/diagram.html

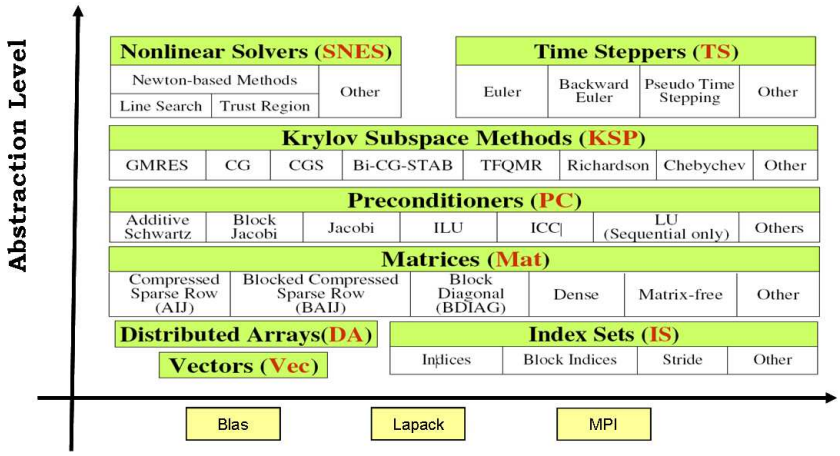


Fig. 1. Hierarchical structure of PETSc

WE show performance results related to the execution time and speed-up obtained on a test case.

Discretization of the problem described in section II is performed using a mesh of $15 \times 15 \times 15$ elements while the particle is discretized by using 128 elements (see figure 2 for mesh representation). This discretization leads to a sparse and symmetric matrix A_0 (see figure 3-(a) for sparsity pattern) with a sparsity pattern of $.03\%$ respet to $O\left(10^{12}\right)$ entries.

This problem has a medium-large level of complexity (the storage of double precision floating point numbers not-zero entries of A_0 requires more than $4GB$ of memory).

We adopt the *row-block data distribution*, which is the standard data distribution adopted in PETSc. In figure 3-(b) we show the *block-row distribution* of A_0 on 4 processors. As expected, such data distribution does not take into account the sparsity pattern of data and it may lead to workload unbalance.

Tests are carried out on a multi processor system with the following configuration [2]:

Hardware configuration: a blade type system composed by:

- 16 blade Dell PowerEdge M600 with,
  - two quad core Intel Xeon E5410@2.33GHz processors,
  - 16 Gb of RAM,
  - A Mellanox Technologies MT25418 Infiniband card,
- A Cisco M SFS7000 switch for internal Infiniband connectivity,

Software configuration: a cluster for parallel application based on:

- Scientific Linux 4.6 operating system,
- Mellanox driver and software for Infini-

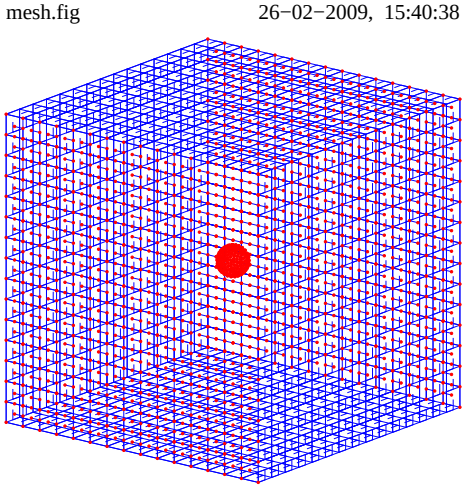
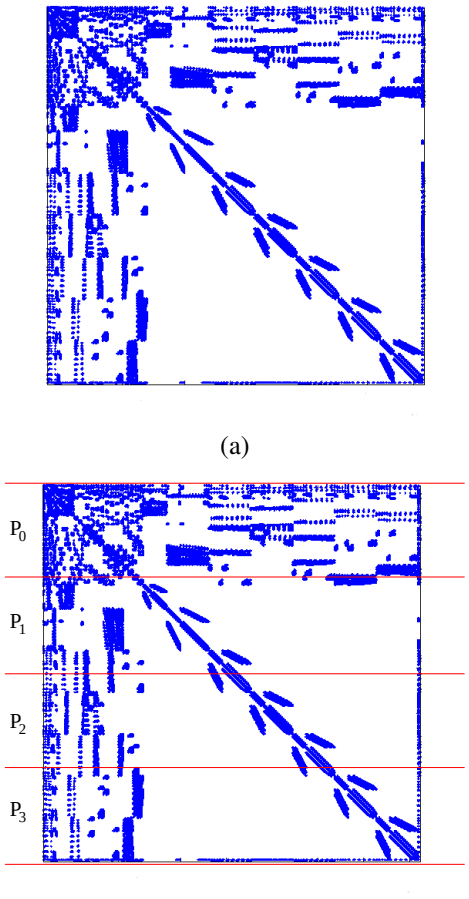


Fig. 2. Representation of discretization mesh

[2] http://scopedma-wn.dma.unina.it



(a)

(b)

Fig. 3. Sparsity pattern of matrix A_0 (a), representation of *block-row* type distribution of A_0 (b)

band connectivity,

- SCoPE Toolkit 1.0 [9](PETSc, ScaLapack, etc.), Intel compilers and libraries.

In figure 4 we show the execution time (a) and speed-up (b) versus the processor number. It is worth to underline that these preliminary experiments focus on the discretization step only. More precisely, we monitor the performance gain obtained introducing concurrency during the discretization step only. As a consequence, the execution time and the speedup refer to point 2 only and not to the entire execution of Algorithm 2.

We appreciate a significative improvement with respect to the mono processor software. Observe that the sequential version of TFEM was not able to perform this computation. As expected, as the processor number grows, the speedup line highlights the quite limited obtained performance gain, because of the limited amount of parallelism that we have introduced. Moreover the data distribution does not take into account the sparsity pattern of the matrix. Of course, a more suitable data distribution will improve computational workload.

V. Future work

The primary intent of this work has been the exploitation of high performance computing resources in order to reduce the overall computational cost required for the simulation of 3D fluido-dynamic processes. We are currently working on the management of parallel objects (matrices and vectors) in TFEM *"parallel"* version, and on the selection of the most efficient linear solver (preconditioned Krylov subspace methods, based on B-Jacobi, ASM, or AMG preconditioners, or LU-based direct methods and Multifrontal methods). Finally, we will be devoted to the visualization and interpretation of results, and comparison to expectations (theories, experiments, or other simulations).

References

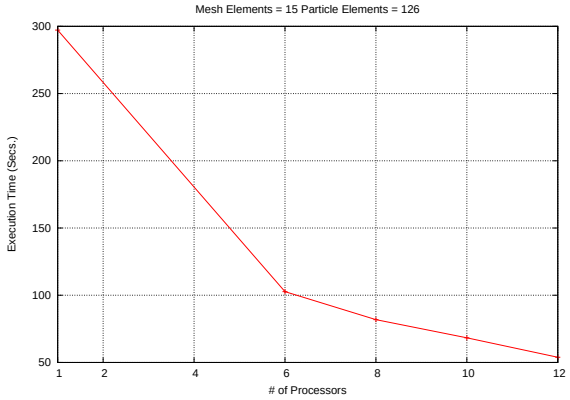
[1] S. Balay, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. Curfman McInnes, B. Smith, H. Zhang, *PETSc Users Manual*, Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory, 2004.

[2] G. D'Avino, M.A. Hulsen, F. Snijkers, J. Vermant, F. Greco, P.L. Maffettone, *Rotation of a sphere in a viscoelastic liquid subjected to shear flow. Part I: Numerical results*, J. Rheol. 52(6), 2008.

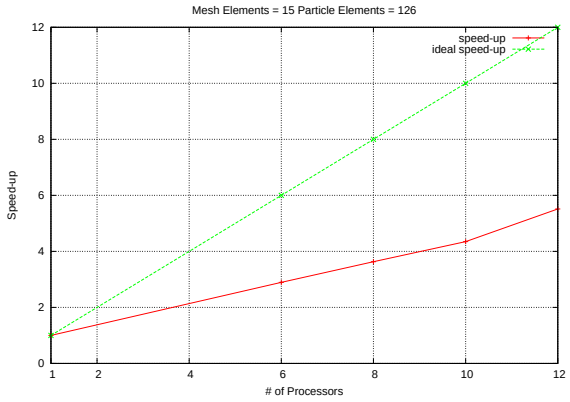
[3] M.A. Hulsen, *TFEM: A toolkit for the finite element method*, Userguide, 2009.

[4] R. Glowinski, T.-W. Pan, T.I. Hesla, D.D. Joseph, *A distributed Lagrangian multipliers/fictitious domain method for particulate flows*, Int. J. Multiphase Flow 25, 1999.

[5] W.R. Hwang, M.A. Hulsen, H.E.H. Meijer, *Direct simulation of particle suspensions in sliding bi-periodic frames*, J. Comput. Phys. 194, 2004.



(a)



(b)

Fig. 4. Preliminary performance results: execution time (a) and speed-up (b) versus the processors number

[6] A.N. Brooks, T.J.R. Hughes, *Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations*, Comp. Meth. Appl. Mech. Eng., 32, 1982.

[7] M.A. Hulsen, R. Fattal, R. Kupferman, *Flow of viscoelastic fluids past a cylinder at high Deborah number: stabilized simulations using matrix logarithms*, J. Non-Newtonian Fluid Mech., 127, 2005.

[8] R. Guénette, M. Fortin, *A new mixed finite element method for computing viscoelastic flows*, J. Non-Newtonian Fluid Mech., 60, 1995.

[9] G.B. Barone, D Bottalico, V Boccia, L Carracciuolo, S Pardi, M Scognamiglio, F Serio, *Middleware Applicativo: lo S.Co.P.E. Toolkit*, Proceedings of Italian National Conference of e-Science, 2008.

A Grid Computational Approach to Two-Phase Flow in Microfluidics

F. Schembri[1], F. Sapuppo[1], E. Leggio[2], M. Iacono-Manno[2], M. Bucolo[1] and L. Fortuna[1]

[1] *DIEES Università degli Studi di Catania, Catania, Italy, florinda.schembri@diees.unict.it*
[2] *Consorzio Cometa (INFN), Catania, Italy*

***Abstract*— In this work the spatial and temporal nonlinear dynamics in microfluidic two-phase flow phenomena are investigated. A numerical model of two phase air/water flows in the serpentine micromixer is implemented exploiting Fluent-Grid technology for investigation on bubbles formation and on important spatial fluid dynamics variables such as the velocity field.**

***Index Terms*— Microfluidics, Two Phase Flow, Micromixer, Computational Fluid Dynamics (CFD), Parallel Processing, Grid, Fluent.**

I. INTRODUCTION

The characterization and control of fluids and particles motion in microfluidic systems, either in vitro artificial devices or in vivo biological systems, are active areas in various research fields; in particular in the biotechnology research [1] for the development of integrated devices for the analysis of biological fluids and particles (Lab-on-Chip).

The research environment related to microfluidic systems involves a variety of processes occurring at the micrometric scales. They can be due to a continuous or discrete flow of fluids, gasses, particles, droplets, emulsions, suspension, etc. In order for them to be functional, the complex physics of such phenomena needs to be investigated.

In particular the quantification and comparison of the nonlinear dynamics involved in formation processes and chaotic advection [2] [3] [4] [5] of such droplets in a microfluidic mixer is not a trivial matter, since experimental observations are not practical, due to the micrometric scale of the devices and require ad hoc opto-electronic instrumentations [5] [6].

Numerical modeling of a two-phase (air/water) flow in a serpentine micromixer, is here considered. Computational Fluid Dynamic (CFD) solution are exploited, allowing a detailed and quantitative characterization of spatial fluid dynamics as the droplets formation process. Because of the two phase flow numerical simulations were very time consuming on a single calculator due to the hardware restrictions a spatial domain parallelization has been necessary.

In this work Fluent 6.3.26® (a well parallelized CFD software) has been used on a Grid infrastructure with the advantage of reducing the computing time by mesh and data partitioning on 48 subdomains.

II. MATERIALS AND METHODS

A. Experimental Setup

The serpentine mixer (SMS0104, ThinXXS), here considered, belongs to the class of passive snake mixer with two inlets.

As shown in Fig. 1a, air and water are inputted through the two inlets of the micromixer using twin pumps (TPS1304, Thinxxs) and therefore obtaining a two-phase fluid through the Y junction. The input flow rate of both fluids are manually regulated through an electronic pump control (EDP0704, Thinxxs).

Through the use of an electro-optical system implemented ad hoc for microfluidic flow studies [6] has been possible the experimental observation of the bubbles flow phenomenon. Fig 1b shows some frames of a video recorded

during an experiment confirming the bubbles passage in a spot of the micromixer.
Experimental studies of the two phase flow dynamics in the serpentine mixer have been performed in previous works [4] [5].

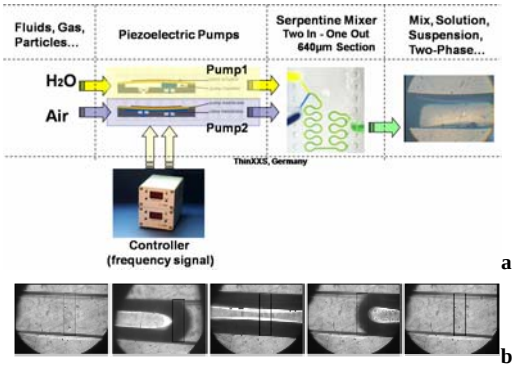


Fig. 1. (a) Microfluidic Experimental Setup (b) Frames related to air bubbles passage in the serpentine.

B. Numerical Model on Grid Infrastructure

The Fluent-Grid environment has been chosen to model the two phase flow process (air-water) in the serpentine micromixer. Fluent can be exploited for parallel processing involving the use of the message passing interface (MPI) and mesh partitioning. Thus Fluent software is adaptable to run on Grid architecture with an InfiniBand interconnection network.
The modeling process of the two phase flow in the serpentine involves several steps: from geometry creation to codes development for the submission on Grid of the parallel CFD jobs.

1) Micromixer Spatial Domain and Mesh

A fundamental step toward the numerical simulation has been the 3D CAD of the solution domain. The serpentine mixer is 121 mm long between confluence and outlet and it has a 640 µm squared section, internal and external radius of curvatures respectively of 320 µm and 960 µm. The serpentine geometry has been created through the software Autocad® and SolidWorks®. A 3D view of the serpentine CAD is show in Fig. 2.

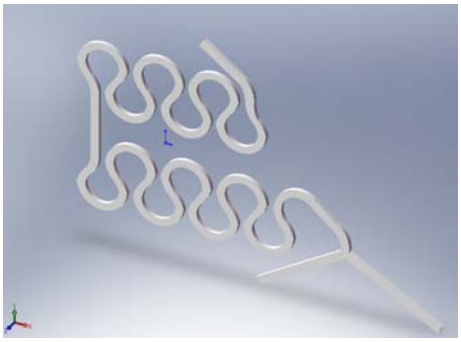


Fig.2. Solution domain.

The mesh creation is particularly delicate because it determines the accuracy of the flow pattern simulations. An ad hoc mesh of micromixer has been set up by means of the software Beta CAE ANSA®.
In the case here considered 302.820 triangular elements have been chosen as superficial mesh of the micro-serpentine, while an unstructured CFD tetrahedral mesh of 2.724.041 elements as the volumetric mesh. In Fig. 3 an hollow into the serpentine mesh shows the internal tetrahedral mesh.

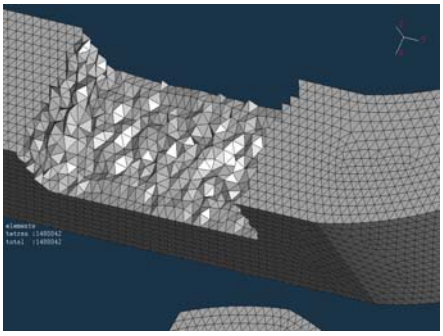


Fig. 3. The superficial and volumetric mesh used in the model.

After mesh definition, the boundaries have been differenced (air inlet, water inlet, etc..) by adding an identification number.

2) Numerical Model

The numerical model involves the Navier Stokes equations to model the laminar flow, and the direct numerical Volume of Fluid (VOF) method to model the free surface of the two immiscible fluids (water and air) in the

microchannel [7] by solving a single set of momentum equations and tracking the volume fraction of each of the fluids throughout the domain. In the VOF method, the fluid location is recorded by employing a volume-of-fluid function which is defined as unity within one fluid regions and zero elsewhere [8].

Constant velocities with linear profile have been applied at the two inlets: 0.05 m/s for water and 0.1 m/s for air. These input velocities determine bubbles formation. The other boundary conditions have been set as: pressure outlet, wetted and not wetted walls.

According to Courant number limitations the time step has been fixed to 0.0002. The Courant number is the ratio of a time step value to a cell residence time leading the numerical stability of the algorithm.

3) Implementation on Grid

For submit the microfluidic job on Grid a Fluent *Journal file* [8] has been created mainly to read the case and data files of the model [8] and set automatically the time step value and the number of iterations in the successive Fluent launch on Grid.

Several files have been necessary to solve the microfluidic two phase flow model on Grid [9]. One of the most important is the bash script file called *mpi.pre.sh* [9] [10]**.** It is an input file created in order to submit and control the Fluent job on the worker nodes (WN). In particular the *mpi.pre.sh* script has the objective of preparing machine file for Fluent, launch other scrip to support the job, copy on the WN the Fluent case and data files, launch Fluent waiting for job results. Thanks to this script the results (images, case-data files) are automatically uploaded in the storage elements (SE).

Another fundamental file to job submission is the JDL [9]. In particular, this file allows to set the number of cores, required for the simulation, on the Computing Element (CE) [9].

III. Results and Discussion

A. Numerical Algorithm Performance on Grid

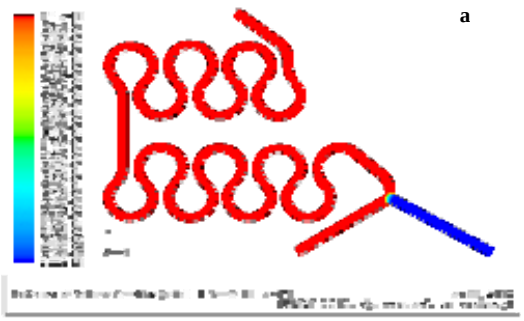
In the case here considered, 48 cores have been chosen to speed up the simulation process taking 22 sec for one time step compared to 5 minutes needed on a Intel Core2 Quad, 2.4GHz. The processed iterations have been 800.000, having in total 8 sec of simulation.

The simulation output amounts to about 50GB of data that has been downloaded on the user interface (UI) space [9] and then processed.

Several variables and parameters could be extracted from the simulation results such as the air and water contours, velocity field, stream line flux etc. Some significant results are shown hereafter: the process of bubbles formation, the velocity field and streamline flux.

B. Flow Pattern Formation

One of the most significant results is certainly the process of pattern formation. This process is considered nonlinear because involves a flux instability between the two phases.



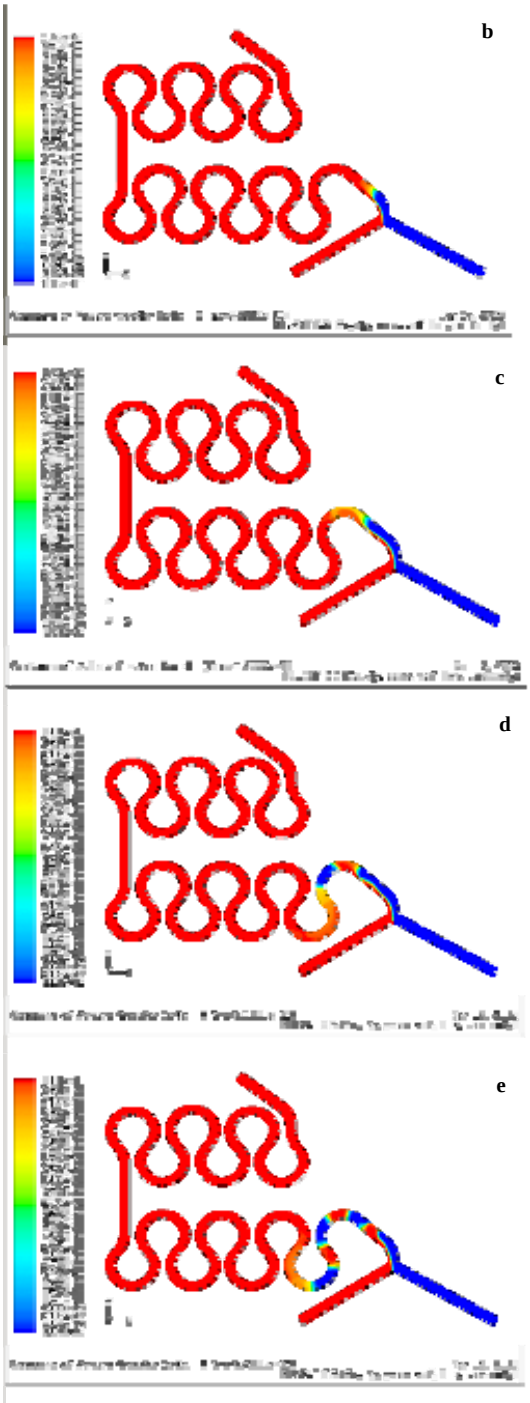


Fig. 4.Coutour of air fraction at time (a)0.3 s(b)0.36 s (c)0.42 s(d)0.51 s (e)0.57 s (f)1.6 s.

Fig. 4 shows some snapshots of the air contours at different times. The image in Fig. 4f in particular shows the periodic flow pattern of the bubbles flow.

C. Velocity Field

Velocity vector of the x-velocity component variation is shown at time of 0.7 s (Fig. 5) in order to analyze the velocity variation due to the bubbles flow in the serpentine. There are different zones visible in Fig. 5 of velocity changes distributed in the serpentine curves.

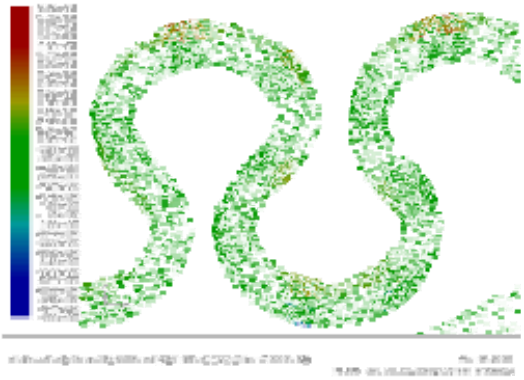


Fig. 5.Velocity vector of x-component velocity variation at 0.7s.

This study is important in order to define the difference in mixing regions distributed along the serpentine.

D. *Streamline Flux*

In order to study the trajectory of the velocity field in the serpentine mixer a streamline representation has been chosen.
Trajectories at different velocities coming side by side have been representing in Fig. 6a,b. It is interesting to notice that the passage of bubbles (Fig. 6c) causes perturbations in the streamline flux and consequently in the velocity field.

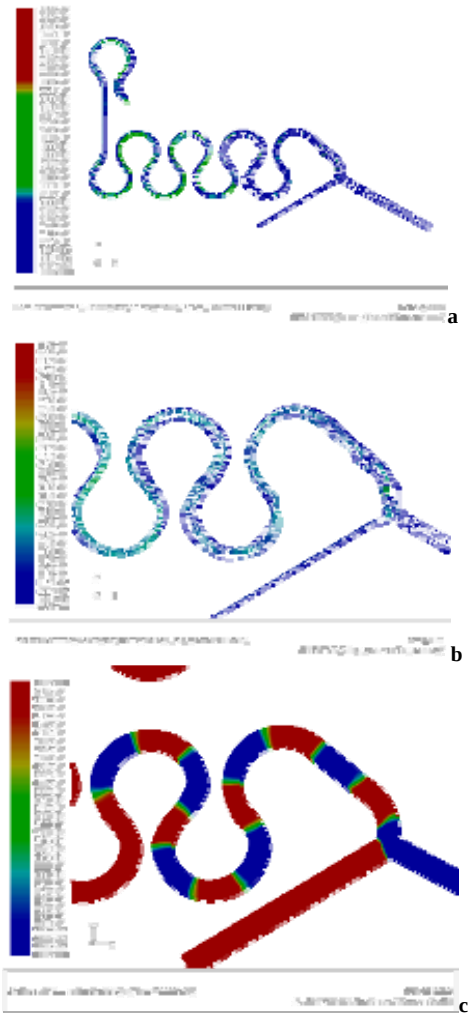


Fig. 6. Streamline flux colored by velocity magnitude at 0.7 s (a) complete serpentine domain (b) detail at the first curves (c) air fraction contour .

IV. Conclusions

This work presents a first approach to time consuming CFD simulations involving the analysis of two phase flow on a complex micro fluidic channel. A Fluent-Grid environment has been exploited considerably improving the time performance and allowing a complex analysis on a 3D domain. Dynamic air fraction and 3D velocity fields of the two phase were extracted as output in order to characterize the mixing regions.

References

[1] R. Barry, I. Dimitri, "Microfluidics in biotechnology", Journal of Nanobiotechnology, 2004, 2:2.
[2] V. Hessel, L. Holger, and F. Schonfeld, "Micromixer-a review on passive and active mixing principles", Elsevier, Chemical Engineering Science, 2005, 60, 2479-2501.
[3] M. R. Bringer, C. J. Gerdts, H. Song, J. D. Tice, and R. F Ismagilov,. "Microfluidic systems for chemical kinetics that rely on chaotic mixing in droplets", Phylos. Trans. Royal Society (London), 2004, 362, 1087-1104.
[4] M. Bucolo, L. Fortuna, F. Sapuppo, F. Schembri, "Chaotic Dynamics in Microfluidic Experiments", The 18th Int. Symposium on Mathematical Theory of Networks and Systems (MTNS 2008), Blacksburg, Virginia, USA, 2008.
[5] M. Bucolo, L. Fortuna, F. Sapuppo, and F. Schembri, "Integrated devices for investigation of nonlinear dynamics in microfluidics", The 10th Experimental Chaos Conference ECC10 (2008), Catania, Italy, 2008.
[6] F. Sapuppo, G. Cantelli, L. Fortuna, P. Arena and M. Bucolo, "Real-Time Monitoring System for Microfluidics", SPIE Europe Microtechnologies for the New Millennium, Maspalomas, Gran Canaria, Spain , 2007.
[7] V. Cristini and Yung-Chieh, Y., "Theory and numerical simulation of droplet dynamics in complex flows-a review", Lab on a Chip, 2004, 4, 257-264.
[8] Fluent 6.3.26 Documentation.
[9] https://grid.ct.infn.it/twiki/bin/view/PI2S2/UserTutorials
[10] http://www.pluto.it/files/ildp/guide/abs/index.html

Florinda Schembri was born in Catania, Italy in 1981. She received the Master Degree in computer science engineering from Catania University in 2006. She worked for one year in a computer science company (Infracom s.r.l) in the field of innovative image processing algorithms. She is from November 2007 a Ph.D. student in Electronics, Automation and Control of Complex Systems at the University of Catania. Her research interests include microfluidic systems analysis and control, Computational Fluid Dynamics applied to microfluidics, nonlinear data analysis, and biomedical image processing. She is IEEE student member.

Francesca Sapuppo currently conducts research, as a postdoctoral associate, at the Department of Electric, Electronic and Systems Engineering at the University of Catania and she is visiting professor at the Department of Electronic Engineering at the University of Messina. She received the M.S. degree in microelectronics engineering in 2003 and Ph.D. degree in electronic and control engineering in 2007 all from the University of Catania, Italy. In 2004 she carried out research at the Microhemodynamic Laboratory of the University of California, San Diego, as visiting scholar and research assistant, starting her work on real-time technologies for microcirculation applications. Currently, her research includes bio-microfluidic control and system engineering, in particular advanced electro-optical instrumentation, polymeric optical interfaces and microsystems, nonlinear technologies as cellular neural networks (CNN) applied to distributed real time monitoring and control of microfluidic processes. Her research includes also nonlinear data analysis and chaos theory applied to biological and microfluidic phenomena.

Emanuele Leggio was born in Ragusa, Italy in 1978. He receveid the Master Degree in Mechanical Engineering from Catania University in 2007. He worked for Consortium SCIRE, Numidia s.r.l., Honor Center of Italian Universities (H^2CU) in the field of Computational Fluid Dynamics (CFD) and Grid computing. On 2008 he joined the Grid team at INFN Catania section developing CFD modeling of Marmore Waterfalls and preparing Fluent-Grid environment.

Marcello Iacono-Manno after Physics graduation during 1990, worked at the Laboratorio Nazionale del Sud, Istituto Nazionale di Fisica Nucleare (INFN), on Data Acquisition and Control Systems developing sequential and parallel C codes for the readout programs and electronic device control. He also developed an off-line application for data analysis based on neural network simulation. After Ph.D. graduation, during 2005 he joined the Grid team at the INFN Catania Section, collaborating to the Trigrid and PI2S2 projects as application supporter focusing on parallel and massive calculus.

Maide Bucolo received the Computer Science Engineering degree from the University of Catania in 1997 and the Ph.D. degree in Electronic and Control Engineering in 2001. She is currently Assistant Professor at the University of Catania. Moreover she is involved in several projects supported by public institutions and consortium of private companies. Her research activity mainly regards study of complex dynamics, cellular neural networks and nonlinear time series analysis for biomedical and economic applications. She is IEEE member.

Luigi Fortuna is full professor of system theory at the University of Catania, where he is Dean of the Engineering Faculty and coordinator of the Ph.D. course in Electronics, Automation and Control of Complex Systems. He is author of more than 400 scientific publications. He is author of ten industrial patents. He is in charge of contracts with public and private companies, such as exMURST, CNR, ENEA, EURATOM, ERG Petroli, ASI, and STMicroelectronics. He is responsible for the FIRB project, nationally—"Cellular self-organizing nets and chaotic nonlinear dynamics to model and control complex systems." He is a IEEE Fellow, and chair of the IEEE Committee on CNNs.

Unsteady Separation and Navier-Stokes Solutions at High Reynolds Numbers

F. Gargano[1,2], A.M. Greco[1], M. Sammartino[1] and V. Sciacca[1]
[1]*Consorzio COMETA, Via S.Sofia 64, I-95123 Catania*
[2]*Dept. of Mathematics, University of Palermo, Via Archirafi 34, 90123 Palermo, Italy*
{*gargano,greco,marco,sciacca*}*@math.unipa.it*

***Abstract*—We compute the numerical solutions for Navier-Stokes and Prandtl's equations in the case of a uniform bidimensional flow past an impulsively started disk. The numerical approximation is based on a spectral methods implemented in a Grid environment. We investigate the relationship between the phenomena of unsteady separation of the flow and the exponential decay of the Fourier spectrum of the solutions. We show that Prandtl's solution develops a separation singularity in a finite time. Navier-Stokes solutions are computed over a range of Reynolds numbers from 3000 to 50000. We show that the appearance of large gradients of the pressure in the streamwise direction, reveals that the viscous-inviscid interaction between the boundary layer flow and the outer flow starts before the singularity time for Prandtl's equation. We observe a relationship between the formation of large gradients of pressure and the various stage of unsteady separation with the loss of exponential decay of the spectrum of the solutions.**

***Index Terms*—Unsteady Separation; Boundary Layer Theory; Navier-Stokes Solutions; Complex Singularities; Spectral Methods.**

I. Introduction

In this paper we shall present the numerical results of an investigation on the phenomenon of unsteady separation in the case of uniform flow past an impulsively started disk. In particular we want to make a comparison between Prandtl and and Navier-Stokes solutions at different Re numbers in the range between 3000 and 50000. We use highly accurate numerical schemes ensuring spectral convergence. In fact we use Fourier spectral $\tau-$method for Prandtl's equation, and Fourier-Chebychev collocation for Navier-Stokes equations. Spectral accuracy is required as we want to analyze the qualitative behavior of the spectrum of Fourier modes of solution. In particular we want to compare the algebraic and exponential decay of the spectrum between Prandtl and Navier Stokes solutions for the different Re numbers. The algebraic and exponential decay of the spectrum are respectively related to the type and to the distance from real axis of complex singularities of solutions. This methodology, called complex tracking singularity, was originally introduced in [13] to investigate the possibility of the finite time blow-up for the 3D Euler's equation. One of the main results in this paper is that the exponential decay of spectrum of Prandtl's solution vanishes in a finite time, in agreement with the seminal work of Van Dommelen and Shen [9], which clarified how separation in Prandtl's equations is linked to the formation of a finite blow-up of the solution in a finite time.

More problematic is the analysis of the relationship between the process of boundary layer separation and the behavior of the spectrum of the solutions. Although the qualitative picture of the Navier-Stokes solutions, with the appearance of high gradients in the streamwise direction, resembles the structure of the singularity in Prandtl's solution, a more careful study reveals that the viscous-inviscid interaction between the

boundary layer flow and the outer flow starts before the singularity time for Prandtl's equations (see [3], [14], [7], [6]). One of the main results of this work is that the formation of large gradients of the pressure on the disk in the various stage of unsteady separation is related to the loss of exponential decay of the spectrum of the solution. Moreover after Prandtl's singularity time the exponential decay seems to stabilize at a value depending on Re number.

II. The initial data and numerical method

A. Statement of the problem

We consider the case of a 2D disk impulsively started in a uniform steady background flow ([11]). The initial datum is therefore the initial solution of the Euler equation, which, in terms of streamfunction, is given by:

$$\psi_E(\theta, r) = U(r - \frac{a^2}{r})\sin(\theta), \qquad (1)$$

where (θ, r) are cilyndrical coordinates defined in the domain $[0, 2\pi] \times [1, \infty]$, U is the streamwise velocity component at infinity, and a is the radius of the disk. Velocity component (u, v) and vorticity ω are obtained by relations

$$u = \frac{1}{r}\partial_r \psi_E, \quad v = -\partial_\theta \psi_E, \qquad (2)$$

$$\omega = -\Delta_{\theta,r}\psi_E, \qquad (3)$$

where $\Delta_{\theta,r}$ is laplacian in cilyndrical coordinates. The problem is solved in vorticity-streamfunction formulation. As $\psi \to \infty$ for $r \to \infty$ we need to truncate the normal physical domain to a finite value R_{max}. This value is chosen big enough to ensure that vorticity here remains negligible for all computational time (let us say $\omega \leq 10^{-16}$). We pass to non-dimensional variables using the diameter of the disk and the velocity at infinity U, so that the Reynolds number is defined here as $Re = \frac{2aU}{\nu}$ where ν is viscosity. The normal physical domain $[1, R_{max}]$ is mapped onto the computational domain [-1,1] by using the domain transformation function

$$\Gamma(r) = \hat{r} = \\ = 1 + \frac{4}{\pi}\arctan(b\tan(\frac{\pi(r - R_{max})}{2(R_{max} - 1)})), \qquad (4)$$

where $b > 0$ is the parameter determining the degree of focusing of the grid points close to the boundary where a better resolution is required as vorticity is generated and large gradients form. Using this transformation the vorticity-streamfunction system to solve in the domain $[0, 2\pi] \times [-1, 1]$ is:

$$\partial_t \omega = A(\hat{r})\partial_\theta \omega + B(\hat{r})\partial_{\hat{r}}\omega + \\ + C(\hat{r})\partial_{\theta\theta}\omega + D(\hat{r})\partial_{\hat{r}\hat{r}}\omega \qquad (5)$$

$$E(\hat{r})\partial_{\theta\theta}\psi + F(\hat{r})\partial_{\hat{r}\hat{r}}\psi + G(\hat{r})\partial_{\hat{r}}\psi = -\omega \qquad (6)$$

$$H(\hat{r})\partial_{\hat{r}}\psi = u, \quad \frac{1}{\Gamma^{-1}(\hat{r})}\partial_\theta \psi = -v \qquad (7)$$

$$\omega(\theta, -1, t) = \xi(\theta, t), \quad \omega(\theta, 1, t) = 0 \qquad (8)$$

$$\psi(\theta, -1, t) = 0,$$

$$\psi(\theta, 1, t) = \psi_E(\theta, R_{max}, t) \qquad (9)$$

$$u(\theta, -1, t) = v(\theta, -1, t) = 0 \qquad (10)$$

$$\omega(\theta, \hat{r}, 0) = 0 \qquad (11)$$

where

$$A(\hat{r}) = \frac{1}{\Gamma^{-1}(\hat{r})}u$$

$$B(\hat{r}) = \frac{1}{Re}\Gamma_{rr}(\hat{r}) - \Gamma_r(\hat{r})v$$

$$C(\hat{r}) = \frac{1}{Re}\frac{1}{(\Gamma^{-1}(\hat{r}))^2}, D(\hat{r}) = \frac{1}{Re}\Gamma_r(\hat{r})^2$$

$$E(\hat{r}) = \frac{1}{(\Gamma^{-1}(\hat{r}))^2}, F(\hat{r}) = \Gamma_r(\hat{r})^2$$

$$G(\hat{r}) = \frac{1}{\Gamma^{-1}(\hat{r})}(\Gamma_{rr}(\hat{r}) + \Gamma_r(\hat{r}))$$

$$H(\hat{r}) = \Gamma_r(\hat{r})$$

The boundary condition for the vorticity on the disk must be evaluated for each time step, as it changes according to the no-slip condition $u(\theta, -1, t) = 0$ for the angular velocity component. In the following section we explain how to recover correct boundary condition through the influence matrix method.

B. Numerical Scheme for Navier-Stokes equations

The numerical scheme we propose to solve the system of equations (5)-(11) is a fully spectral method. To accomplish this, first we approximate the solution in the $\theta-$variable through Fourier series, employing periodicity in θ. Then we choose as grid points in the computational normal domain [-1,1] the Gauss-Lobatto points, given by $\hat{r_k} = \cos(k\pi/N)_{k=0,..N}$, where N+1 are the number of grid points. The corresponding effect in the physical domain, applying the inverse of the transformation (4), is a focus in the point $r = 1$ and $r = R_{max}$, more pronounced in $r = 1$ as b decreases. Now we are able to use to use the approximation:

$$\omega(\theta, \hat{r}, t) \approx \sum_{k=-M/2}^{k=M/2} \omega_k(\hat{r}, t) e^{ik\theta}, \qquad (12)$$

where M are the grid point in the domain $[0, 2\pi]$, and ω_k are the Fourier modes, depending on the normal variable $\hat{r}$ and time t. Using Galerkin-Fourier method ([2], [18]) we obtain for each mode the system

$$\partial_t \omega_k = N(u_k, v_k, w_k) + \frac{1}{Re} L(w_k), \quad (13)$$

$$F(\hat{r})\partial_{\hat{r}\hat{r}}\psi_k + G(\hat{r})\partial_{\hat{r}}\psi_k - E(\hat{r})k^2\psi_k = -\omega_k \quad (14)$$

$$H(\hat{r})\partial_{\hat{r}}\psi_k = u_k, \quad \frac{ik}{\Gamma^{-1}(\hat{r})}\psi_k = -v_k, \quad (15)$$

where N is the convective term, and L is the diffusive one. As the system is solved in the Gauss-Lobatto points, Chebychev-Collocation method is used to ensure a fully spectral convergence (see [2], [18] for a complete review of this method and convergence properties).

The temporal discretization used is the following Adam-Bashforth - Implicit Backward differentiation (AD/BDI2)

$$\frac{3\omega_k^{n+1} - 4\omega_k^n - \omega_k^{n-1}}{2\Delta t} = N^{n+1} + \frac{1}{Re} L^{n+1} \qquad (16)$$

This is a two-step semi-implicit method, as the diffusive term L is treated implicitly, and the convective non linear term N is approximated by means of an Adams-Bashworth extrapolations by $N^{n+1} \cong 2N^n - N^{n-1}$. The product are evaluated in both direction using pseudo-spectral approximation with anti-aliasing procedure. To find vorticity at the boundary we use the Influence Matrix Method, which is based on the superposition of particular solutions solving time-dependent Stokes problems. Introducing the operators

$$\Phi = F(\hat{r})\partial_{\hat{r}\hat{r}} + G(\hat{r})\partial_{\hat{r}} - \frac{k^2}{r^2} - \frac{3Re}{2\Delta t}, \qquad (17)$$

$$\Upsilon = F(\hat{r})\partial_{\hat{r}\hat{r}} + G(\hat{r})\partial_{\hat{r}} - \frac{k^2}{r^2} \qquad (18)$$

for any fixed k the Fourier modes $(\omega_k(\hat{r}), \psi_k(\hat{r}))$ can be sought as solution of the Stokes-type problem

$$\Phi\omega_k = N, \quad \Upsilon\psi = -\omega_k, \qquad (19)$$

$$\omega_k(-1) = \xi_k, \quad \omega_k(1) = 0, \qquad (20)$$

$$\psi_k(-1) = 0, \quad \psi_k(1) = p_k. \qquad (21)$$

where ξ_k is the unknown vorticity at the boundary, and p_k is the value of ψ_k in 1. To find ξ_k the solution is decomposed in

$$\omega_k = \tilde{\omega_k} + \xi_k\bar{\omega}_k, \quad \psi_k = \tilde{\psi_k} + \xi_k\bar{\psi}_k, \qquad (22)$$

where the pair $(\tilde{\omega}_k, \tilde{\psi}_k)$ solves

$$\Phi\tilde{\omega}_k = N, \quad \Upsilon\tilde{\psi}_k = -\tilde{\omega}_k, \qquad (23)$$

$$\tilde{\omega}_k(-1) = 0, \quad \tilde{\omega}_k(1) = 0, \qquad (24)$$

$$\tilde{\psi}_k(-1) = 0, \quad \tilde{\psi}_k(1) = p_k. \qquad (25)$$

and the pair $(\bar{\omega}_k, \bar{\psi}_k)$ solves

$$\Phi\bar{\omega}_k = N, \quad \Upsilon\bar{\psi}_k = -\bar{\omega}_k, \qquad (26)$$

$$\bar{\omega}_k(-1) = 1, \quad \bar{\omega}_k(1) = 0, \qquad (27)$$

$$\bar{\psi}_k(-1) = 0, \quad \bar{\psi}_k(1) = 0. \qquad (28)$$

At each step only $(\tilde{\omega_k}, \tilde{\psi_k})$ are evaluated as $(\bar{\omega_k}, \bar{\psi_k})$ does not depend on time and can be evaluated in the preprocessing stage. The vorticity ξ_k is then determined by imposing the no-slip condition at the boundary

$$\xi_k\bar{\psi}_k' + \tilde{\psi}_k' = 0. \qquad (29)$$

The mode (ω_k, ψ_k) are then recovered by (22), and used to find velocity component (u_k, v_k)

through (15).
We have computed the solution for Reynolds number $3{\cdot}10^3, 10^4, 2{\cdot}10^4, 5{\cdot}10^4$, and we choose the number of grid points and the value of R_{max} according the value of Re, as shown in the following table

Re	GRID	R_{max}	dt
$3 \cdot 10^3$	512×513	5	$5 \cdot 10^{-5}$
10^4	512×513	6	$2.5 \cdot 10^{-5}$
$2 \cdot 10^4$	1024×513	4	$7.5 \cdot 10^{-6}$
$5 \cdot 10^4$	4096×513	2	$5 \cdot 10^{-6}$

The parameter b in the transformation function (4) has been chosen for all computation equal to 0.1. For all Re numbers the numerical simulations starts with coarser grid, which is increasingly refined when the flow develops small scale structures.

C. Prandtl's equations and numerical scheme

To describe the behavior of the flow close to the boundary in the limit case $Re \to \infty$, one must consider Prandtl's equations . These equations can be obtained as asymptotic limit of Navier-Stokes equations assuming that viscous force are not negligible only close the rigid boundary. In the case of impulsively started disk the system to solve is:

$$\partial_t u + u\partial_\theta u + v\partial_r u - U\partial_\theta U = \partial_{rr} u, \quad (30)$$

$$\partial_\theta u + \partial_r v = 0 \quad (31)$$

$$u(\theta, 0, t) = v(\theta, 0, t) = 0 \quad (32)$$

$$u(\theta, \infty, 0) = U \quad (33)$$

$$u(\theta, r, 0) = U\,, \quad (34)$$

where $U = 2\sin(\theta)$ is the velocity of the uniform background flow. The system must be solved in the domain $[0, 2\pi] \times [0, \infty[$ for the velocity component u. The incompressibility condition (31) allows to recover the rescaled normal velocity v through an integration in the normal variable. Equation (32) and Eq.(33) give the boundary condition and the matching condition with the outer flow, while (34) is the initial condition. We solve the system in the computational domain $[0, 2\pi] \times [0, Y]$, where Y is chosen big enough to ensure matching condition (33) (i.e $|u(\theta, Y, t) - U|$ must be small for all computational time). In our calculation a value $Y = 20$ has satisfied the required condition as in [8].
The numerical method we use is the well known τ-method (see [2], [18]), which ensure a spectral convergence of the solution. This is accomplished by considering Fourier-Chebychev expansion of the solution in the new domain $(\theta, \tilde{r}) = [0, 2\pi] \times [-1, 1]$, where $\tilde{r} = \frac{2r-Y}{Y}$, obtaining the approximation

$$u(\theta, \tilde{r}, t) \approx \sum_{k=-M/2}^{k=M/2} \sum_{j=0}^{j=N} u_{kj}(t) e^{ik\theta} T_j(\tilde{r})\,, \quad (35)$$

where T_j are the Chebychev polynomials of the first kind. Introducing the variable ζ defined as $\tilde{r} = \cos(\zeta)$, we write the above expression as:

$$u(\theta, \zeta, t) \approx \sum_{k=-M/2}^{k=M/2} \sum_{j=0}^{j=N} u_{kj}(t) e^{ikx} \cos(j\zeta)\,. \quad (36)$$

Temporal scheme used is a semi-implicit two stage Runge-Kutta scheme, where the diffusion term is treated by Crank-Nicholson implicit approximation, and the non linear term is treated explicitilly. We have compute Prandtl's solution in a grid of 2048×2048 points, and the adaptive time step chosen to ensure stability in our computation varies according to the CFL stability condition $\Delta t = \frac{5}{2Mmax|v|}$.

III. Singularity tracking

In this section we give a description of the method we use to track a singularity in the complex plane for an analytic function. The technique is used to track the singularity for both Navier-Stokes and Prandtl's solutions.
If an analytic function $u(z)$ has an algebraic singularity at the complex location $z^* = x^* + i\delta$, and if at the singularity the behavior is $u(z) \approx (z - z^*)^\alpha$, then the Fourier spectrum of $u(z)$ has the following asymptotic behavior [carr,pers]:

$$u_k \sim |k|^{-(1+\alpha)} \exp(-\delta|k|) \exp(ix^*k) \quad (37)$$

Therefore if one estimates the rate δ of the exponential decay of the spectrum one gets the distance of the complex singularity from the real axis; the estimate of the period of the oscillations of the spectrum gives the real location x^* of the singularity. Resolving the rate of algebraic decay $1+\alpha$, one can classify the singularity type. In particular, when one deals with a PDE with solution $u(t)$, one can detect, with a good degree of reliability, the time at which the singularity develops as given by the first t_c at which $\delta(t_c)=0$.

It is possible to extend the above technique to bi-variate functions, and we refer to [13], [16] for more details. Let $u(x_1,x_2)$ a function with Fourier expansion:

$$u(x_1,x_2)=\sum_{k_1,k_2} u_{k_1k_2}e^{ik_1x_1}e^{ik_2x_2}\ .$$

If in the Fourier **k**–space one considers those modes (k_1,k_2) such that $k_1=k\cos(\theta)$, $k_2=k\sin(\theta)$ where $k=|(k_1,k_2)|$, then the Fourier coefficients, when $k\to\infty$, have the following asymptotic behavior:

$$u_{k_1k_2}\approx k^{-(\hat{\alpha}(\theta)+1)}e^{-\hat{\delta}(\theta)k}e^{ikx^*(\theta)}$$
$$(k_1,k_2)=k(\cos(\theta),\sin(\theta))\ . \qquad (38)$$

The width of the analyticity strip $\overline{\delta}$ is the minimum over all directions θ, i.e. $\delta^*=\min_{\hat{\theta}}\delta(\theta)$.

A second way to extend the singularity tracking method to bi-variate functions is to define the shell-summed Fourier amplitudes:

$$A_K\equiv\sum_{K\le|(k_1,k_2)|<K+1}|u_{k_1k_2}|, \qquad (39)$$

which is a kind of discrete angle average of the Fourier coefficients. The asymptotic behavior of these amplitudes is:

$$A_K\approx CK^{-(\alpha+1)}\exp(-\delta K), \qquad (40)$$

when $K\to\infty$, where δ gives the width of the analyticity strip while the algebraic prefactor α gives informations on the nature of the singularity.

As pointed out in [16], using a steepest descent argument, one can see that the two techniques are equivalent. In fact, if one denotes with θ^* the angle where $\hat{\delta}(\theta)$ takes its minimum (i.e. $\delta^*=\hat{\delta}(\theta^*)$), one has that $\delta=\hat{\delta}(\theta^*)$ and that $\alpha=\hat{\alpha}(\theta^*)$.

This method can be applied to every Fourier expansion, as in the case of Eq. (36), and then apply the singularity tracking method to the correspective shell-summed amplitude.

IV. Results

In this section we analyze the results obtained by the singularity tracking method, described in previous section, to the shell-summed Fourier amplitudes applied to both Navier Stokes solutions (5)-(11) for various Reynolds, and Prandtl's solution (30)-(34), expressed in Fourier-Chebychev expansion.

As the Fourier-Chebychev modes of Navier-Stokes solutions vary as normal physical domain changes, we need to evaluate the solutions in a common domain, to avoid any possible discrepancies in the results. We choose as common normal physical domain $\Lambda=[1,2]$, as we have seen that for $r>2$ vorticity remains always negligible of all Re numbers. The point λ_i in Λ are exactly the linear mapping of Gauss-Lobatto's points κ_i in Λ, by $\lambda_i=(\kappa_i/2+1.5)$. We easily map the solution in Λ as is always possible to recover the physical value of a function in any point of the domain from Fourier-Chebychev modes obtained from (36) by applying Clenshaw's reccurence formula. In this case a simple interpolation is to be avoided because the interpolation-error in the result could affect the spectral convergence of the solution.

A. Boundary Layer Results

Is well known that Prandtl's solution forms a real singularity at time 1.5, as pointed aut in the seminal work of Van Dommelen and Shen ([9]). The singularity tracking method is able to recover this result. In Fig.1 one can see the loss of exponential decay of shell-summed amplitude up to singularity times. Fitting these results with a function of the form of (40), we track the behavior in time of the strip of analicity $\delta_{Sh}(t)$ and the algebraic pre-factor $\alpha_{Sh}(t)$

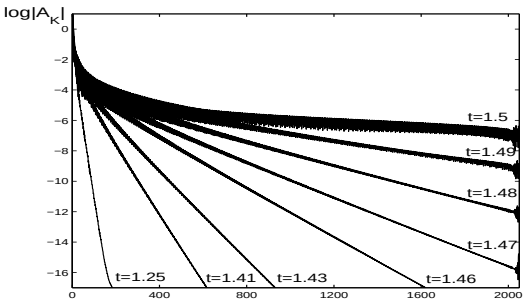


Fig. 1. The behavior in time of shell-summed amplitudes for Prandtl's solution up singularity time in lin-log coordinates.

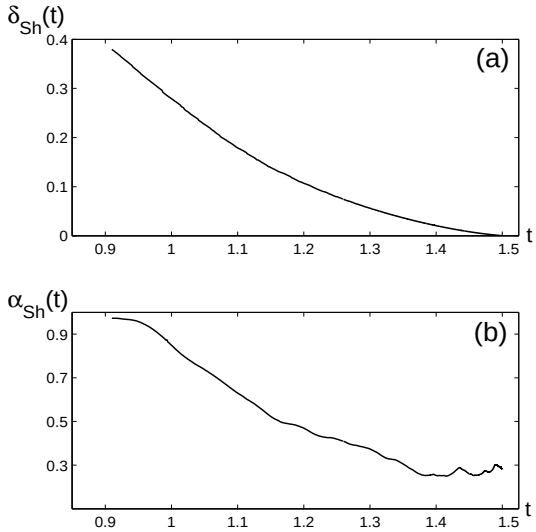


Fig. 2. Singularity formation for Prandtl's solution. At time 1.5 the width of analicity strip vanish as a cubic root singularity forms.

(Fig.2), obtaining that the strip of analicity δ_{Sh} shrinks to zero at time 1.5 as a consequence of the formation of a cubic root singularity. Physically this means that vorticity is ejected in the outer flow from within the boundary layer, and normal velocity component becomes infinite in boundary layer scale. These physical behavior is quite similar to other Prandtl's solution blow-up for different initial data ([3], [7], [17]),

B. Navier Stokes Results

We now show the results obtained from Navier-Stokes computations, for all Re numbers considered. Applying the singularity tracking method we obtain that the distance δ_{NS} of the closest singularity to the real axes (see Fig.3) slightly increases up to time $t \approx 1.1 - 1.15$, where a local maximum in time of $\delta_{NS}(t)$ forms. This situation seems to be independent from the different values of Re numbers. At this time there is the formation of a local maximum and a local minimum also in the streamwise pressure gradients at the boundary (defined as $\partial_\theta p_b = \Gamma_r(\hat{r})\partial_{\hat{r}}\omega|_{\hat{r}=-1}$). These critical points forms in the range $2 \leq \theta \leq 2.5$. This quantity does not change in boundary layer theory, while in NS solutions varies according to the viscid-inviscid interaction that accelerates the unsteady separation ([3], [14], [7], [6]). In Fig.4 is shown the temporal evolution in time of $\partial_\theta p_{\hat{r}=-1}$ starting at $t = 1$ until $t = 1.5$ with increments of 0.1 for all Re numbers. One can see that after the formation of the local maximum a kink starts to form, more pronounced for higher Re numbers. This reflects the fact that a recirculation region has formed close to the boundary, and grows in time creating a pressure variation. It is evident the correspondence in the variation of δ_{NS}: a rapid decreasing behavior is visible in Fig.3 for all Re after $t = 1.1$, as a consequence of the complex singularity getting close to real axes. The main differences between the various Re (as one could expect) is that as Re increases δ_{NS} decreases. At Prandtl's singularity time δ_{NS} still shows a decreasing behavior, and seems to stabilize at approximatively $t \approx 1.9$ to a value varying according the Re number. At this time several corotating eddies have developed, and the pressure gradient on the disk shows several kinks giving evidence of the possibility that solutions have developed other complex singularities.

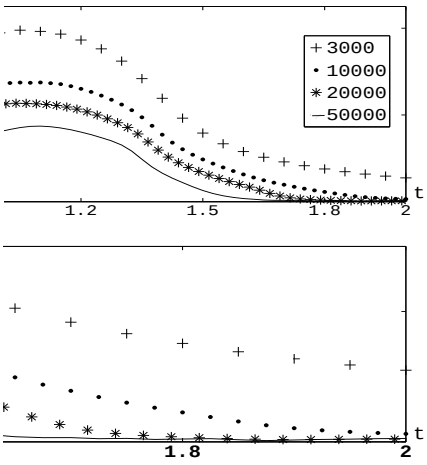


Fig. 3. The behavior of $\delta_{NS}(t)$ for various Re numbers, and a zoom after time 1.6

V. Conclusion

We have computed the solutions of Prandtl's and Navier-Stokes solutions at different Re numbers (3000-50000) with fully spectral numerical schemes. We approximate the solutions in Fourier-Chebychev expansion in the computational domain $[0, 2\pi] \times [-1, 1]$, and we apply the singularity tracking method to the shell summed amplitudes of the Fourier coefficients. We are able to track in time the width of analicity strip δ, and the results show that Prandtl's solution forms a cubic root singularity at time 1.5, which reveals the unsteady separation precess. For NS solutions we obtain that the width of analicity strip δ_{NS} grows up to time 1.1 forming a maximum in time for all Re numbers we consider. Then it starts to decrease and a couple of local minimum-maximum forms in the streamwise pressure gradient on the disk $\partial_\theta p_b$. We argue that the decreasing behavior in time of δ_{NS} is bound to the physical formation of large gradients in the pressure. At Prandtl's singularity time δ_{NS} get close to zero as Re increases, and reach a local minimum in time at $t \approx 1.9$, when seems to stabilize to a distance

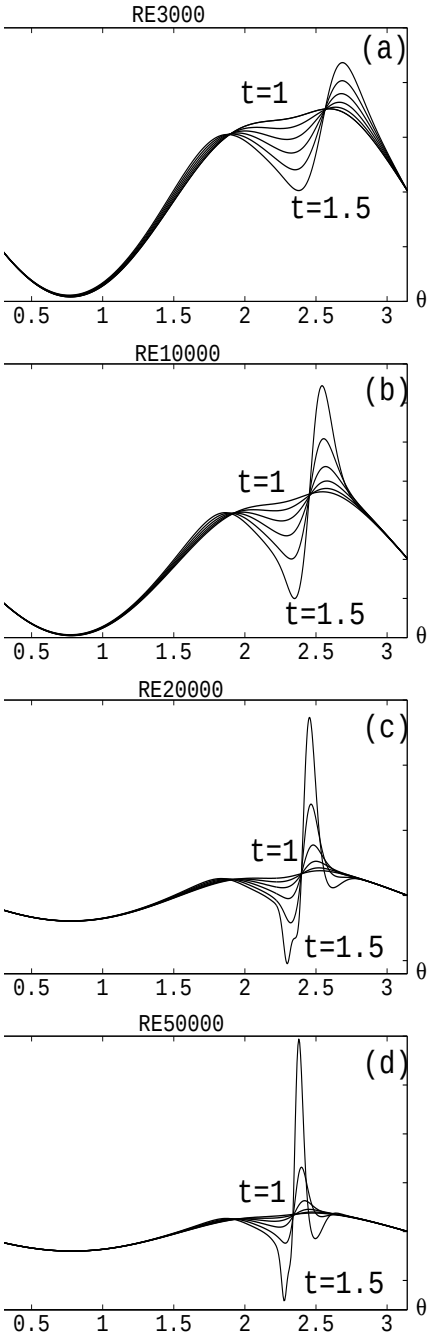


Fig. 4. The evolution in time of streamwise pressure gradient on the disk start at time $t = 1$ up to $t = 1.5$ (increments of 0.1) for various Reynolds number.

depending on the Re number.

ACKNOWLEDGMENT

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006).

REFERENCES

[1] C. Bardos S. Benachour, M. Zerner, *Analyticite des solutions periodiques de l'equation d'Euler en deux dimensions*, C.R. Acad. Sci. Paris 282A, 995-998 (1976).

[2] C. Canuto, M.Y. Hussaini, A. Quarteroni, T. Zang, Spectral Methods in Fluid Dynamics, 2th ed., Springer Verlag, Berlin Heidelberg, 1988.

[3] K.W. Cassel, A comparison of Navier-Stokes solutions with the theoretical description of unsteady separation, *Phil. Trans. R. Soc. Lond. A* **358** 3207–3227 (2000).

[4] S.J. Cowley, Laminar Boundary-Layer Thoery: A 20th Century Paradox?, *Proceedings of ICTAM 2000*, eds. H. Aref and J.W. Phillips, (Kluwer, 2001).

[5] G. Della Rocca, M.C. Lombardo, M. Sammartino, V. Sciacca, Singularity tracking for Camassa-Holm and Prandtl's equations, *Appl. Numer. Math.* **56** 1108–1122 (2006).

[6] G. Della Rocca, F. Gargano, M. Sammartino, V. Sciacca, Transition to turbulence and singularity in Boundary Layer Theory, *Wascom 2007*, Word SCI.PUBL., 199-209 (2008)

[7] F. Gargano, A.M Greco, M. Sammartino, V. Sciacca, Transition to turbulence ans Singularity in Boundary Layer Theory, *Grid Open Days 2007* **1** 245–253 (2008).

[8] L. Hong and J.K. Hunter, Singularity formation and instability in the unsteady inviscid and viscous Prandtl equations, *Comm. Math. Sci.* **1(2)**, 293–316 (2003).

[9] L.L. van Dommelen, S.F. Shen, The Spontaneous Generation of the Singularity in a Separating Laminar Boundary Layer, *J. Comp. Phys.* **38** 125–140 (1980).

[10] R. Krasny, A study of singularity formation in a vortex sheet by the point vortex approximation, *J. Fluid. Mech.* **167** 292–213 (1986).

[11] O. Lamb, Hydrodynamics, sixth edition Cambridge University Press, Cambridge, 1932.

[12] M.C. Lombardo, M. Cannone, M. Sammartino, Well-Posedness of the Boundary Layer Equations *SIAM J. Math. Anal.* **35** 987–1004 (2003).

[13] T. Matsumoto, J. Bec, and U. Frisch, The analytic structure of 2D Euler flow at short times.*Fluid Dyn. Res.* 36(4-6) (2005) 221–237.

[14] A.V. Obabko, K.W. Cassel, Navier-Stokes solutions of unsteady separation induced by a vortex, *J. Fluid Mech.* **465** 99–130 (2002).

[15] O.A. Oleinik, V.N. Samokhin, Mathematical Models in Boundary Layer Theory, Chapman & Hall/CRC, Boca Raton, FL, 1999.

[16] W. Pauls, T. Matsumoto, U. Frisch, and J. Bec, *Nature of Complex Singularities for the 2D Euler Equation*, Physica D, **219** , 40–59 (2006).

[17] V.J. Peridier, F.T. Smith, J.D.A. Walker, Vortex induced boundary-layer separation. Part 1. The unsteady limit problem $Re \to \infty$ *J. Fluid Mech.* **232** 99-165 (1991).

[18] R. Peyret,Spectral Methods for Incompressible Viscous Flow, Springer Verlag, New York, 2002.

[19] M.C. Pugh and M.J. Shelley, Singularity formation in thin jets with surface tension, *Comm. Pure Appl. Math.*, **51(7)** 733–795 (1998)

[20] M. Sammartino, R.E. Caflisch, Zero viscosity limit for analityc solutions of the Navier-Stokes equation on a half space. I: Existence for Euler and Prandtl equations, *Comm. Math. Phys.*, **192** 433-461 (1998).

[21] M. Sammartino, R.E. Caflisch, Zero viscosity limit for analityc solutions of the Navier-Stokes equation on a half space. II: Construction of the Navier-Stokes solution, *Comm. Math. Phys.*, **192** 463-491 (1998).

[22] M. Sammartino, F. Gargano, M.C. Lombardo, V. Sciacca, Singularity Formation and separation Phenomena in Boundary Layer Theory, *Workshop On Partial Differential Equations and Fluid Mechanics.*,Warwick, Lect.Notes in Math,Springer (accepted).

[23] M.J. Shelley, A study of singularity formation in vortex–sheet motion by a spectrally accurate vortex method, *J. Fluid. Mech.* **244** 493–526 (1992).

[24] C. Sulem, P.-L. Sulem, and H. Frisch, Tracing complex singularities with spectral methods, *J. Comput. Phys.* **50** 138-161 (1983).

Enabling Applications on gLite Based Grid Using Virtual Machines

G. Minutoli[1],A. Puliafito[2]
[1]University of Messina, Engineering Faculty, Messina, Italy, gminutoli@unime.it
[2]Consorzio COMETA, Catania, Italy, apuliafito@unime.it

***Abstract*—Predominantly used in scientific contexts, Grid Computing imposes costraints related to underlying infrastructure, forcing in many cases users to rethink or rewrite their applications. But, developing or porting a complex application into a Grid environment represents often a challenging task: numerous technical and practical issues have to be taken in duly consideration and many applications are commercial codes for which source is not available.**

Inspired by Amazon's EC2 model, this paper presents a virtualization based system (*GliteVM*) that - ensuring QoS, scalability, high-availability and security - allows users to execute any existing application in a gLite based Grid infrastructure without changing their current work setting and with a limited effort.

***Index Terms*—Grid, Virtual Machines, VirtualBox, Xmpp, gLite.**

I. Introduction

Based on Internet the "World Wide Web", or simply the "Web" as ordinary it is called, lets people to share the universe of network-accessible information distributed on millions of computers around the world. Similarly Grid Computing allows the shared and coordinated use of heterogeneous resources in dynamic virtual organizations: namely set of individuals and/or institutions which, defined by sharing rules, do not exchange files but share direct access to computers, software, data, and other resources geographically distributed and belonging to distinct administrative domains [1].

Nowadays, systems based on this paradigm are largely used in scientific and business contexts in order to execute applications mainly related to engineering, Earth science, bioinformatic, chemistry, physics and astrophysics. Fluid dynamic algorithms, simulations of aerodynamics problems, software performance analysis tools and thermodynamic or electromagnetic simulations represent some examples of the most popular applications executed on Grid, making optimal use of parallel processing capabilities, with huge benefits in terms of time and performance.

Nevertheless, although Grid Computing offers great potential for executing applications, its practical deployment is strongly constrained by the fact that many existing applications are legacy or compiled against specific versions of libraries that exist on a certain version of operating system: whereupon they can not be directly used into Grid environments [2]. The solution should be to simply rewrite such applications. However, developing or porting a complex application into a Grid environment represents often a challenging task: numerous technical and practical issues have to be taken in duly consideration and many applications are commercial codes for which source is not available. Moreover, although this might be possible, only a few users can take the time, effort and money to do it making sure that their application runs correctly.

Inspired by Amazon's EC2 model [3], we propose a system that, through the adoption of virtualization, enables the users of a gLite based Grid infrastructure to execute any already existing applications with a limited effort and without changing their current work setting.

The proposed solution, called *GliteVM*, allows to create on-demand virtual working environments that, customized in terms of software and resources according to specific users' requirements, can be used by the Grid users to run their applications. In other words, they can run their applications on Grid adapting the runtime environment to the application instead of porting the application to the runtime environment [4]. Once created, the virtual environments will be accessible to the users by a wide array of devices (PC, Laptop, Thin Client) using common standard protocols (RDP, HTTP, SSH) in full compliance with the consolidated operating modes of the users.

The rest of this paper is organized as follow: in section II an overview of the state-of-the-art is provided. Section III clarifies functionalities and objectives of the proposed solution describing some real use cases. Section IV discusses technical details about its implementation within a gLite based Grid infrastructure, while some experimental measurements are presented in section V. Concluding remarks are given in section VI.

II. Related Works

The use of virtualization within Grid environments has been widely investigated. It represents a powerful way to solve - in a fast and simple manner - some of the typical issues found in current large-scale distributed systems: *security, resources control, user applications support*.

In the context of the SVGrid project [5], for example, Zhao et al. propose a virtualization based solution to protect filesystem and computers of a Grid infrastructure from possible malicious attacks. They deploy one or more virtual environments in order to isolate user applications, whose requests to access the network or the filesystem are mediated by a dedicated monitor virtual environment. In this manner, they prevent that dangerous grid applications modify sensitive files (such as system binaries or configuration), install backdoors, inject virus to disrupt the normal execution or - by abusing the network - use the victim grid resource as a midway to attack other hosts [5]. In the In-Vigo [6], instead, authors explore the potentiality of virtualization technologies in the context of the Grid resources management, highlighting how to create virtual Grid infrastructure where application services are instantiated on-demand. Similarly, Ruth et al [7] presented Violin, an infrastructure that integrates virtual environments with virtual networks to provide virtual distributed environments.

In terms of customization, Keahey et al. have introduced for the first time the concept of *Virtual Workspace*: a configurable virtual execution environment that can be customized according to user's requirements, in terms both of software and resource provision [8][9]. A concept subsequently extended by the same authors in order to emcopass the notion of cluster that matches clients software and hardware requirements [10]. In [11] and [2], the authors describe the implementation of transparent systems whereby the user of a Grid infrastructure can submit his applications and - with minimum extra effort - execute them within customized guest environments. A Grid service called VMPLant is presented in [12]. It implements automated configuration and creation mechanisms of flexible virtual execution environments that, once configured to meet user application needs, can be dynamically instantiated to provide homogeneous execution environments across distributed Grid resources.

Numerous research projects investigated the potentiality of using virtualization within of Grid systems. Many of them were conducted exploiting the middleware, such as in Globus [13] or LCG [14]. Little or nothing, instead, seems to be reported in literature about the integration of virtual environments within the gLite middleware [15]. In this vision, our *GliteVM* solution may be considered a possible starting point for future research works aimed at clarifying how the benefits derived from the use of virtualization can be offered to the users of a gLite infrastructure.

III. System Architecture

The key idea of the *GliteVM* system is to use virtualization techniques within a gLite-based Grid infrastructure in order to allow users to create and manage on-demand customized virtual environments, where their applications are executed without any chance (see Fig. 1).

Composed of clusters of virtual machines - previously created and personalized in terms of memory, disk space, operating system and software - these virtual environments are presented to the user as dedicated servers where he has the same degree of access of an administrator. Furthermore, he is offered the possibility to interact by a wide array of devices (PC, Laptop, Thin Client) using common standard protocols (RDP, HTTP, SSH). Methodologies such as virtual private network (VPN), based on X.509 certificates and the SSL protocol, are also exploited in order to create secure communication between the user's device and the virtual execution environments, ensuring data confidentiality, message integrity and prevention of replay attacks.

In general, two main activities referred respectively as *"Create Virtual Machine"* and *"Instance Virtual Machine"* can be identified. In the former phase, the user creates a virtual disk image of the complete software environment he needs. The image, so produced, is then stored on the storage elements and adequately registered on the file catalog of the Grid infrastructure. During the *"Instance Virtual Machine"* phase, instead, the file with the virtual disk image is recovered from a Grid node and exploited to create a virtual machine, with the characteristics (such as ram, network, processing core number) specified by the user.

Similarly to what happens in the Amazon's EC2 model and in other systems described in the section II, the real magic in *GliteVM* is that users can create and destroy virtual environments at will. Nevertheless, differently from other solutions for Grid systems, our proposal is characterized by a job-oriented architecture that does not require any modifications of the adopted Grid middleware. The only requirement to be met is the pre-installation of the selected virtualization software on all the computational resources involved in the Grid infrastructure. In such a way, all the resources offered by the overall Grid system can be indiscriminately used with huge benefits in terms of scalability.

Finally, it is also important to highlight how the functionalities of *GliteVM* are not related to a specific user application or to a particular operating system. Virtual execution environments with Linux or Microsoft operating systems can be easily created to execute a wide range of applications, independently from the research field. In order to further clarify these concepts, a real use case concerning the opmitized use of wireless systems with simulation is presented in the following.

A. Use Case

Electromagnetic simulation has dramatically improved the design of a wide range of wireless systems by making it possible for researchers to simulate their operation and predict their performance without building and testing complex and expensive prototypes.

In this vision, software simulators have become an important instrument for scientists in the research work. Produced and distributed by Remcom [16], *Wireless InSite* is a powerful electromagnetic modeling tool for predicting the effects of buildings and terrain on the propagation of electromagnetic waves (Fig. 2). It models the physical characteristics of the rough terrain and urban building features, performs the electromagnetic calculations, and then evaluates the signal propagation characteristics.

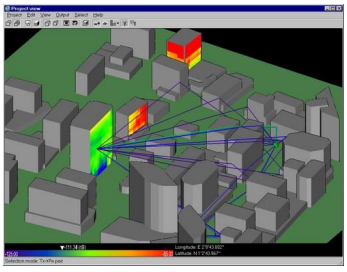


Fig. 2. One screenshot of the Wireless InSite software.

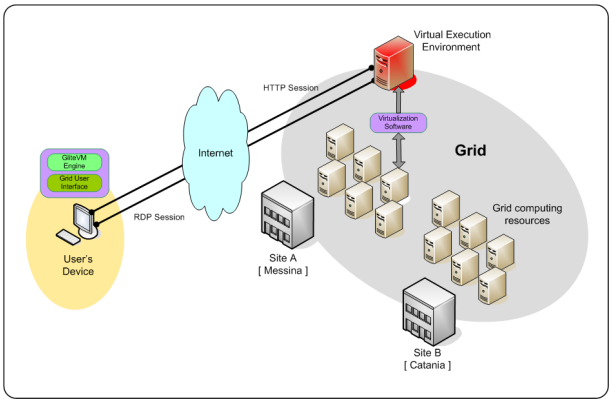


Fig. 1. Example of single virtual execution environment created with *GliteVM*.

At the time Wireless InSite is only available for the Windows 2000 and XP operating systems. Thus, it can not be directly executed into a Grid environment that is mainly based on Linux. For these reasons, the use of this software simulator - to analyse how the locations of the transmitters and receivers within an urban area affect the propagation of electromagnetic waves - was chosen as use case in order to show the functionalities and test the efficiency of the proposed *GliteVM* system.

Initially, a virtual disk image of a software environment with Windows XP and the Wireless InSite tool has been created and stored on the storage system of the Grid infrastructure. Then, it has been used to instance, on the Grid nodes, one or more custom virtual execution environments with which the researcher interacts using a set of Remote Desktop sessions, and where he can run his simulations without changing his current work setting, but with an optimal use of processing capabilities offered by a Grid system.

IV. Implementation

The development of *GliteVM* has been carried out using the gLite based Grid testbed built by Consorzio COMETA within the PI2S2 Project running at the Messina University [17]. Different modules were implemented. They exploit Java classes and Bash shell scripts for the graphical user interfaces, the system logic, the interaction with Grid and virtualization software.

As shown in Fig. 3, the core of the proposed system is so called *GliteVM Engine*. Running on the user's device, it is a multithreading Java application that implements the system basic logic, enabling a user to create and manage the virtual execution environments he needs. Developed according to a 3-tier model, it consists of three main components:

- ***Graphical User Interface*** (GUI): that addresses all the issues concerned to the interaction between user and system, offering the former a classic graphical interface to create and customize his virtual environments;
- ***Server Engine***: that exploits the services of the middleware environment called *Grid User Interface* (installed also on the user's device) to manage all the user's requests submitted by the GUI;
- ***Local Database***: where all information about the state and the configuration of the different virtual execution environments, requested by user, are stored.

In order to clarify the functionalities of the various elements, a simple workflow example is briefly described in the following.

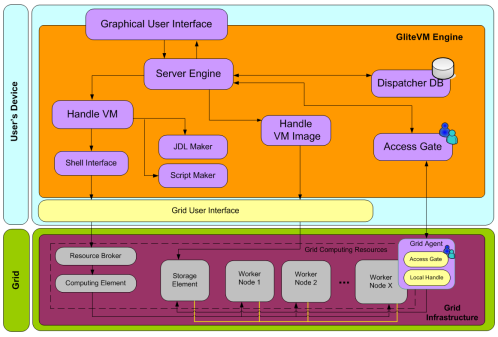


Fig. 3. Architecture of the *GliteVM* system.

A. Workflow Example

The use of the *GliteVM* system includes an initial phase during which the user creates and personalize a virtual disk image of the desired software environment using the provided tool. In particular, during the development of *GliteVM*, the VirtualBox tool [18] has been selected to create virtual environments in users' devices and to instance and manage them within the computers involved in the Grid infrastructure. Thanks to the easy installation process, open source license and the ability to run unmodified guest operating systems (Windows, Linux and so on) with only small performance losses, it represents an ideal solution.

Once the image file is created, it is recorded on the Storage Elements(SEs) of the Grid system by the *Server Engine* component, exploiting functionalities of the *Handle VM Image* module. In order to enable its correct recover during the *"Instance Virtual Machine"* phase, the image of each single user is recorded on the LFC Catalog of the Grid system under the logical path */grid/VO_NAME/GliteVM*. More precisely, for each user, the *GliteVM* system creates a directory whose name is an unambiguous ID. Such directory is then used to create symbolic links to all the images of the virtual disks created by a specific user and stored on the SEs.

Whenever the user wishes to run a virtual execution environment on the Grid, he submits his own request by the GUI offered by the *GliteVM Engine* application, specifying the virtual disk image to be used and other parameters such as ram, number of processors, network configuration and any port to be redirected. The user's request is then analysed and managed by the *Server Engine* that, through the *Handle VM* module, starts the activation procedure of the requested virtual environment. It submits a Grid job (called for convenience *GliteVM Job*) that will recover the selected virtual disk image, creating the virtual environment with the characteristics specified by the user. In particular, *JDL Maker* and *Scripts Maker* modules are responsible to automatically create the JDL file and other scripts necessary for the correct workflow of this job.

Once the virtual environment is active, the user can interact with it by a wide range of common standard protocols (RDP, HTTP, SSH) in full compliance with his consolidated operating mode. Nevertheless, in this vision, a strong interaction between the *GliteVM Engine* application and the *GliteVM Job* is required. In fact, the former has to know which worker node on the Grid is used for the creation of the virtual environment, and which ports the user can use to interact with it. In the proposed system, the communication between such components takes place using a message oriented middleware based on the Extensible Messaging and Presence Protocol (XMPP) [19]. A point-to-point queueing model has been adopted, implementing *Access Gate* and *Grid Agent* modules in order to enable the *GliteVM Job* to post the requested information. For each created virtual execution environment, one new record is inserted in the local database storing all these information and the configuration selected by the user. In this manner, he will have the opportunity to interact with its virtual environment at any time he wishes.

V. Experimental Results

In terms of user satisfaction, the most important parameter that has been evaluated is the latency time (T_L), expressed as the time interval between the submission of a request by the user (start up of a virtual machine) and the instant

when it is running and ready to perform the required tasks. In accordance equation

$$T_L = T_S + T_R + T_{OS} \quad (1)$$

T_L refers to the time needed for a *GliteVM Job* to run the selected virtual environment. More specifically, it is defined as the sum of the scheduling time (T_S), the recovering time (T_R) and the operating system time (T_{OS}). T_R refers to the time needed to have a job running on a WN. T_R is the time interval needed for a *GliteVM Job* to recover the file housing the virtual machine disk image. T_{OS}, instead, represents the initialization time of the virtual machine.

Since T_{OS} strongly depends on how the user configures (in terms of memory, operating system, storage space, software) the desired virtual execution environment, during the performance analysis only the contributions of T_S and T_R were assessed.

Fig. 4 shows the T_S evolution during an ordinary working day for the PI2S2 Grid infrastructure (for each hour we calculated the average times) highlighting that T_S is not well predictable. Characterized by a standard deviation equal to the 103% of its mean and a maximum value almost nine times the mean itself, it often forces users to wait a long period before resources become available. As discussed in a previous work [20], a solution based on a job reservation strategy by mean of idle_jobs could be adopted in order to overcome this issue, improving also the perceived quality of service.

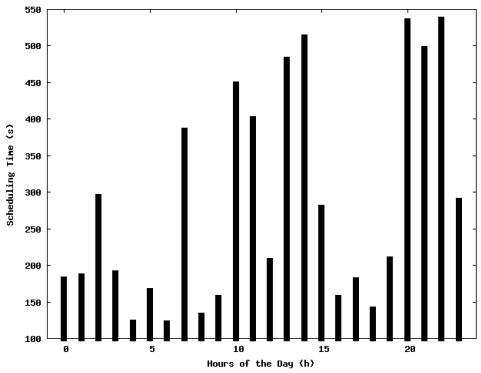


Fig. 4. T_S distribution during a working day in PI2S2 Grid.

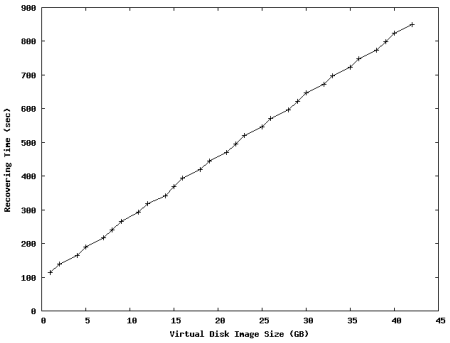


Fig. 5. The variation of T_R versus virtual disk image size

Fig. 5, instead, shows the variations of the recovering time (T_R) versus the size of the file housing the virtual disk image. As expected, the results demonstrate that increasing the size of the file, the recovering time increases in a linear manner with an angular coefficient equals to 20 sec/GByte. Nevertheless, a further reduction of T_L could be obtained keeping T_R as low as possible through storing images file as simple compressed files. Advantages in term of latency time, could also derive by creating more replica of each virtual disk image and by disseminating them in a geographically distributed manner. This allows the Grid nodes, where the virtual environments are running, to recover the image file from the SE closer with the lowest network latency. Finally, a data management solution based on network distributed file systems can be used for fast virtual execution environments instantiation and efficient run-time execution, allowing to read virtual disk image directly from SE and avoiding to transfer the file on the worker nodes [21].

VI. Conclusions

In the last decades, Grid Computing has been widely exploited in scientific and business environments. Nevertheless, it often imposes constraints related to underlying infrastructure, forcing users to rethink their applications and

greatly limiting the practical use of this technology. Only few users can take the time, effort and money to rewrite or port their complex applications on a Grid infrastructure, resolving numerous technical and practical issues that this involves.

In this article the implementation of GliteVM has been presented. It represents a valid example of how the virtualization technologies can be used to enable users to execute any existing software application in a gLite based Grid infrastructure without changing their habit and with a limited effort. Feasibility has been also proved by evaluating the performance of our tool. Finally, we believe that *GliteVM* may represent a possible starting point for future research works aimed at clarifying how benefits derived from the use of virtualization can be offered to users of an infrastructure based on the gLite middleware.

VII. Acknowledgment

This work makes use of results produced by the PI2S2 project managed by the Consorzio COMETA. A project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale Ricerca Scientica, Sviluppo Tecnologico, Alta Formazione (PON 2000-2006). More information is available at www.consorzio-cometa.it and www.pi2s2.it. We also would like to thank Dr. Antonino Greco of the Engineering Faculty of the Reggio Calabria University for the useful support on the topics covered in the use case.

References

[1] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," *International J. Supercomputer Applications*, 2001.

[2] A. Norton, "AutoXen: Automating Virtual Machine Deployment within the Grid," Department of Physics and Astronomy, University of Victoria - Victoria - Canada, Tech. Rep., January 2007.

[3] "Amazon EC2 - http://aws.amazon.com/ec2."

[4] B. Quetier, V. Neri, and F. Cappello, "Selecting a Virtualization System for Grid/P2P Large Scale Emulation," in *Proceedings of the EXPGRID*, 2006.

[5] X. Zhao, K. Borders, and A. Prakash, "SVGrid: A Secure Virtual Environment for Untrusted Grid Applications," in *Proceedings of the 3rd International Workshop on Middleware for Grid Computing*, 2005.

[6] S. Adabala, V. Chadha, and P. C. et al., "From Virtualized Resources to Virtual Computing Grids: The In-Vigo System," *Future Generation Computer Systems*, vol. 21, pp. 896–909, 2005.

[7] P. Ruth, X. Jiang, D. Xu, and S. Goasguen, "Virtual Distributed Environment in a Shared Infrastructure," *Computer*, vol. 38, pp. 39–47, 2005.

[8] K. Keahey, I. Foster, and T. F. X. Zhang, "Virtual Workspaces : Achieving Quality of Service and Quality of Life in the Grid," *Quality of Life in the Grid. Scientic Programming Journal*, 2005.

[9] K. Keahey, I. Foster, T. Freeman, X. Zhang, and D. Galron, "Virtual Workspace in the Grid," *Euro-Par*, pp. 421–431, 2005.

[10] X. Zhang, K. Keahey, I. Foster, and T. Freeman, "Virtual Cluster Workspace for Grid Applications," *TR-ANL/MCS-P1246-0405*, 2005.

[11] A. Garcia, "Uso di Macchine Virtuali (Xen) per Garantire Servizi di Grid," Ph.D. dissertation, Universita degli Studi di Perugia - Facolta di Scienze Matematiche, Fisiche e Naturali - Corso di Laurea in Informatica, 2006.

[12] I. Krsul, A. Ganguly, and J. Z. et al., "Proceedings of the acm/ieee sc2004 conference," *VMPlants: Providing and Managing Virtual Machine Execution Environments for Grid Computing*, 2004.

[13] "Globus Toolkit - http://www.globus.org/toolkit/."

[14] "Lcg Project - http://lcg.web.cern.ch/lcg/."

[15] "gLite Middleware - http://glite.web.cern.ch/glite/."

[16] "Remcom - http://www.remcom.com/."

[17] "Pi2s2 Project - http://www.pi2s2.it."

[18] "Virtualbox - http://www.virtualbox.org/."

[19] "Xmpp - http://xmpp.org/."

[20] G. Iellamo, G. Minutoli, D. Bruneo, and A. Puliafito, "GridVideo: a Grid based Multimedia Application," in *Proceedings 3rd EGEE User Forum*, 2008.

[21] M. Zhao, J. Zhang, and R. Figueiredo, "Distributed file system virtualization techniques supporting on-demand virtual machine environments for grid computing," *Cluster Computing*, vol. 9, pp. 45–56, 2006.

G. Minutoli received the Laurea degree in Electronic Engineering from the University of Messina, Italy, in 2005. Hes currently attending his Ph.D in Advanced Technologies for Information Engineering at the University of Messina. His research interests are in the areas of distributed computing, Grid, virtualization.

A. Puliafito is a full professor of computer engineering at the University of Messina, Italy. His interests include parallel and distributed systems, networking, wireless and GRID computing. He is currently the director of the RFIDLab, a research lab on RFID and wireless, the vice-president of the Consorzio Cometa whose aim is to enhance and exploit high performance computing. He is actively working in grid computing and virtualization in the projects TriGrid VL, PI2S2 and RESERVOIR.

CUDA Based Implementation of Parallelized Pollard's Rho Algorithm for ECDLP

M. Chinnici[1], S. Cuomo[2], M. Laporta[2], A. Pizzirani[2], S. Migliori[3]
[1]*ENEA-FIM-INFOPPQ, Casaccia Research Center, Via Anguillarese 301, 00123 S.Maria di Galeria, Italy, marta.chinnici@enea.it*
[2]*Università degli studi di Napoli FEDERICO II, Dipartimento di Matematica e Applicazioni R.Caccioppoli Via Cinthia, 80136 Napoli, Italy, {salvatore.cuomo, mlaporta}@unina.it, a.pizzirani@gmail.com*
[3]*ENEA-FIM, Enea-Sede, Lungotevere Thaon di Revel n. 76, 00196 Roma, Italy*

***Abstract*—The recent introduction by NVidia of Compute Unified Device Architecture (CUDA) libraries for High Performance Computing on Graphic Processing Units has started the trend of video cards for resolution of many computationally hard problems in different areas like fluid dynamics, molecular dynamics, computer vision and astrophysics. In this paper we show how CUDA libraries and hardware can be introduced in cryptography as a cryptoanalytic tool. Increase of data communications made data cryptography a real necessity of the daily life. Sometimes, private key cryptosystems are enough, more often public key cryptosystems are needed for communication through insecure channels. Cryptosystems based on the theory of the elliptic curves offer schemas with a relatively low communication overhead. In general, security is strongly related to a presumed intractability of an arithmetic problem like Discrete Logarithm Problem (DLP). In particular, testing the security of the elliptic curve cryptography reduces to testing the solvability of the DLP in the group of points of an elliptic curve (ECDLP). Various methods are known to be more or less efficient in solving instances of the general DLP. Some of them have deterministic running time. Those methods with probabilistic running time like Pollard's rho method have a better trade off between space and time. We describe an implementation of a parallelized Pollard's rho attack on ECDLP, based upon recent results about the optimization of Pollard's rho method and enhanced by some "ad-hoc" choices for CUDA.**

***Index Terms*—Cryptoanalysis, CUDA, Elliptic Curves, High Performance Computing.**

I. Introduction

Cryptography is essentially aimed at protecting data from unauthorized access. This is particularly important when data involve sensible informations and are transmitted on insecure channels. Typical examples are business via internet and the payment with a credit card. The data involved in such transactions are usually encrypted to make it harder for an attacker to retrieve secret informations. In the past, the key for encryption was the same for decryption raising a serious problem regarding key distribution. In 1976 W.Diffie and M.Hellman[3] invented an agreement protocol that allows two users to exchange a secret key over an insecure channel without any prior contact. This event is commonly considered the birth of public-key cryptography. Then, relying on some hard mathematical problem, many cryptosystems have been proposed. However, since some attacks to such math problems succeded most of these cryptosystems become insecure or simply impratical. Actually, three mathematical problems are still considered to be hard: the integer factorization problem (IFP), the discrete

logarithm problem in the multiplicative group of a finite field (DLP) and in the group of points of an elliptic curve (ECDLP). There is no real proof that the aforementioned problems are intractable. However, a lot of work has been done to try to solve them efficiently. All these efforts amount to the development of subexponential-time algorithms for IFP and DLP, but none for ECDLP. Elliptic curve cryptography (ECC) became more and more attractive essentially for such a reason. Moreover, parameters of the ECC are usually much smaller than parameters of cryptosystems based on IFP and DLP. Consequently ECC has lower communication overhead.

II. Elliptic curves

Let $\mathbb{K}$ be a field of characteristic $\neq 2, 3$. The set E of points $(x, y) \in \mathbb{K} \times \mathbb{K}$ satisfying the equation $y^2 = x^3 + ax + b$ with $a, b \in \mathbb{K}$, is called *elliptic curve* whenever x^3+ax+b has no multiple roots in $\mathbb{K}$. The definition of an elliptic curve is slightly more complicated in when the charateristic of $\mathbb{K}$ is 2 or 3. The set E, enriched with a so-called "point to infinity" O_∞ and a well defined addition $+$, becomes an *elliptic group,* denoted by $E(\mathbb{K})$ where the point O_∞ acts as the group identity. If $\mathbb{K} = \mathbb{R}$, the real field, then the addition can be described geometrically through the method "chord-tangent"(see [10] p.55).The inverse of a point $P = (x, y)$ is $-P = (x, -y)$ by definition.Moreover, one has the following explicit formulas for the sum and the doubling of points on $E(\mathbb{R})$. If $P = (x_1, y_1)$, $Q = (x_2, y_2)$ and $P + Q = (x_3, y_3)$, then $x_1 \neq x_2$ implies

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2 \quad ,$$

$$y_3 = -y_1 + \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) \quad , \quad (1)$$

while $P = Q$ implies

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1 \quad ,$$

$$y_3 = -y_1 + \left(\frac{3x_1^2 + a}{2y_1}\right)(x_1 - x_3) \quad . \quad (2)$$

More remarkable is the fact that these formulas are still valid in $E(\mathbb{K})$ for a generic ground field $\mathbb{K}$, where the so called *elliptic curve discrete logarithm problem* can be formulated as it follows: *given* $P, Q \in E(\mathbb{K})$, *determine the integer* k *(if there's one) so that* $Q = kP = \underbrace{P + P + ... + P}_{k \text{ times}}$. This problem is significantly hard if the ground field is finite.

Indeed, V. Miller[7] and N. Koblitz[5] proposed (independently each other) to use the elliptic group $E(\mathbb{F}_p)$, defined on a finite field $\mathbb{F}_p$, as an arithmetic base of a cryptosystem.

III. ECDLP

Altough ECDLP is a particular case of DLP, there is no generic algorithm with subexponential running time that solves it. One reason is that there's a primary difference between the underlying algebraic structures, i.e. the multiplicative group $\mathbb{F}_p^*$ of a finite field $\mathbb{F}_p$ for DLP and the elliptic group $E(\mathbb{F}_p)$. Mainly, while $\mathbb{F}_p^*$ is completed in a structure with two operations, the elliptic group has only its own addition. For example, the *index-calculus methods*, which solves DLP instances with subexponential running time, fail when it comes to elliptic groups (except in very special and well-understood cases). As usual, algorithms for ECDLP are classified as it follows: *generic algorithms* which are appliable to all instances of ECDLP, *special algorithms* which take advantage of the particular instance of the problem. Since all special attacks to the ECDLP can be easily avoided by means of a suitable choice of the parameters, it is more interesting to focus on generic algorithms. The most used generic methods are variations of the Pollard's rho method. Indeed, in 1997 CERTICOM introduced a list of ECDLP challenging problems offering a money prize for each solution[1]. The solved problems got a solution through the use of a parallelized Pollard's rho method.

IV. POLLARD'S RHO ALGORITHM FOR ECDLP

Let us consider $P, Q \in E(\mathbb{F}_p)$ and assume that we want to compute k such that $Q = kP$. The main idea of Pollard's rho algorithm is to determine distinct pairs (c', d') and (c'', d'') of integers modulo n such that $c'P + d'Q = c''P + d''Q$ where n is the order of the subgroup $\langle P \rangle$ generated by P. Then, one can compute $(c' - c'')P = (d'' - d')Q = (d'' - d')kP$, which implies $(c' - c'') \equiv (d'' - d')k \pmod{n}$. Thus, $k \equiv (c' - c'')(d'' - d')^{-1} \pmod{n}$. The naive method requires to generate at random $c, d \in [0, n-1]$, compute $cP + dQ$ and store each triple $(c, d, cP + dQ)$ in a table sorted by the third element until a point $cP + dQ$ is obtained twice (this occurrence is called "collision"). The *birthday paradox*[1] helps to estimate the expected number of iterations (or equivalently the complexity of the algorithm) before a collision is found. This number is approximately $\sqrt{\pi n/2} \approx 1.2533\sqrt{n}$. Instead of randomly generated points Pollard[9] proposed an iteration function acting on $\langle P \rangle$ with a pseudo-random behaviour. If this function is "random enough", then, the algorithm has the same expected running time of the naive method. The original Pollard's function partitions $\langle P \rangle$ into three subsets S_1, S_2, S_3 of approximately equal size. Then, from the starting point P_0 it iterates $P_1 = f(P_0), P_2 = f(P_1), ..., P_{i+1} = f(P_i)$. More precisely, it is defined as:

$$f(P_i) = \begin{cases} P_i + a_1 P + b_1 Q & if \quad P_i \in S_1 \\ 2P_i & if \quad P_i \in S_2 \\ P_i + a_2 P + b_2 Q & if \quad P_i \in S_3 \end{cases}$$

Some experimental results by E.Teske[11][12] showed that the running time of the Pollard's algorithm tends to the expected value $\sqrt{\pi n/2}$ as the number of subsets S_i increases. Mainly storage can be efficiently reduced to a negligible amount at the cost of some extra computation. By using Floyd's algorithm called "tortoise and hare"([4] exercises 6 and 7, page 7) it can be reduced to a constant. Van Oorschot and Wiener[13] proposed to record only points satisfying a precise condition (for example, the last 30bits of the x coordinate have to be equal to zero) for a better trade off between space and performances. Moreover, they showed that the algorithm can be efficiently parallelized on an arbitrary number of processors. While each of them generates a random walk from a different starting point, collision detection is completed by another designated computer.

[1]The birthday paradox can be formulated as it follows: how large the number of people must be in a room in order to expect at least two of them have the same birthday ? Surprisingly the number is small: $\sqrt{\pi 365/2} \approx 24$.

V. CUDA

CUDA is a general purpose parallel computing architecture developed by NVidia. Programming of CUDA devices is realized mainly through "C for CUDA", an extension of the C language that gives user access to CUDA capabilities. Even if C is the main language in CUDA hardware programming, third party wrappers are available for Python, Fortran, Java and MatLab. Actually, as it is reported by NVidia, there are millions of CUDA-capable gpus which are already installed with prices ranging from a few euros for hardware with limited computing capabilities (20€-30€for an 8400GS-256mb video card) to thousands of euros for high-end hardware with 4 teraflops (single precision) power (Tesla C1060 with 960 cores). Some advantages offered by CUDA architecture are: scattered reads(code has access to all memory addresses), a fast shared memory region (a region that grants really high performances and can be used by all threads together), full support of integer and bitwise operations and fast downloads and readbacks to and from gpu.

CUDA has also some limitation that must be considered while developing software: no support for recursive functions on the device, division and inversion are computationally expensive and the device memory management is difficult(threads using device memory should

access it avoiding multiple requests on the same bank).

VI. CUDA BASED IMPLEMENTATION OF PARALLELIZED POLLARD'S RHO ALGORITHM

Considering high processing power of actual gpus, it makes sense to take advantage from them for a parallelized Pollard's rho algorithm. As we already said, that integer division and modulo operations are really expensive. Hence, one has to find suitable solutions for an efficient modular arithmetic, especially when we handle multiword integers and we deal with formulas 1 and 2 on an elliptic curve. Moreover, integer words should be accurately written in memory, even because simultaneous accesses of more than one thread to the same bank cause performance losses.

The implementation of the program is described as it follows:

1) The host makes precomputations needed for the Pollard's rho algorithm.
2) Precomputed data is sent to the device.
3) The host starts threads on gpu. Such threads generate pseudorandom points through the iteration function. Then they look for *distinguished points* (having last 30bit of x coordinate all equal to zero) and report them to the host.
4) The distinguished points are stored into a hash table, where the host looks for collisions. The algorithm stops if a collision is found.

Modular arithmetic. Modular addition is operated naively adding single words from the least significative digits and recording a carry from each previous single word addition. Each addition is followed by a subtraction of one modulus if the sum is greater than it.

The modular difference is operated in an analogous way. Here one has to record borrows of the subtraction of each single word. A negative result requires an addition of one modulus.

In our implementation with CUDA, the most delicate part is the modular multiplication via the Montgomery product ([14] p.395-397) with the coarsely integrated operand scanning method (CIOS)[6]. For all the elaborate details we refer the reader to [8].

The iteration function. The starting points are linear combinations of P and Q. Each point is generated with a different multiple of P. If t threads are executed, each of them is associated to a starting point $A_l = a_l P + q_l Q$ where $0 \leq l < t$ and $0 \leq a_l, q_l < n$. Our iteration function is a so-called "add only" function which partitions $\langle P \rangle$ into r subsets. Let $A_{l,i} = (x_{l,i}, y_{l,i})$ be the point corresponding to the walk of the l-th thread and the i-th iteration. The iteration function is defined as: $f(A_{l,i}) = A_{l,i+1} = A_{l,i} + b_j P$ if $x_{l,i} \equiv j (\mathrm{mod}\, r), \quad \forall j = 0, ..., r-1, \quad \forall l = 0, ..., t-1$.

The congruence $x_{l,i} \equiv j (\mathrm{mod}\, r)$ can be easily checked through bitwise operators if r is a power of 2. In our application we choose $r = 64$.

Points representation and storage. The array $(b_0 P, ..., b_j P, ..., b_{r-1} P)$ is stored by using affine coordinates. More precisely, since the size of these coordinates turns out to be small and since the points do not vary within the program, the array can be fitted into the *constant memory* of the graphic card. That allows to avoid memory problems due to simultaneous accesses of more than one thread to the same point. Moreover note that all data of the curve are recorded in the constant memory. All the other points are represented with the so-called Jacobian coordinates([2], 3.2), which allow us to avoid division and inverse calculation when we add points on the elliptic curve. In order to get coalesced memory access, a single word of each Jacobian coordinate is memorized in a locations multiple of 16 (i.e. the size of the bank). In this way, each thread always accesses the same memory bank. By using just a mid-level gpu, an 8800GTS with g92 chip, with a preliminary version of our application we have generated about 320.000 points/seconds on the curve over $\mathbb{F}_{109}$ listed in CERTICOM site.

VII. CONCLUSIONS

CUDA-enabled gpus turn out to form a very interesting platform for high performance computing for many remarkable reasons: the wide diffusion, a rapidly and continuously increasing power, a good performance/price ratio and the possibility of installing more than one gpu into a single workstation.

In this paper we have shown that graphic cards can be useful to improve on performances for ECDLP resolution. We think that all we have done here is new because all CERTICOM problems were attacked only by using cpus (in distributed environment). Now a mixed approach cpu-gpu can also be considered (gpu being a co-processor). At least, computations for our problem can be performed only on gpu while cpu remains free and available for other jobs.

REFERENCES

[1] CERTICOM, *Certicom ECC challenge*, $http://www.certicom.com/$, (1997).

[2] A. J. Menezes, D. Hankerson and S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer-Verlag New York, Inc., 2003.

[3] W. Diffie and M. E. Hellman, *New Directions in Cryptography*, IEEE Transactions on Information Theory, IT-22 (1976), pp. 644–654.

[4] D. E. Knuth, *Art of Computer Programming, Volume 2: Seminumerical Algorithms (3rd Edition)*, Addison-Wesley Professional, 1997.

[5] N. Koblitz, *Elliptic Curve Cryptosystems*, Mathematics of Computation, 48 (1987), pp. 203–209.

[6] C. K. Koc and T. Acar, *Analyzing and Comparing Montgomery multiplication Algorithms*, IEEE Micro, 16 (1996), pp. 26–33.

[7] V. S. Miller, *Use of Elliptic Curves in Cryptography*, Advances in Cryptology − CRYPTO '85: Proceedings, (1986), pp. 417+.

[8] A. Pizzirani, *Il problema del logaritmo discreto sulle curve ellittiche e relazioni con la crittografia*, Tesi di Laurea presso Università degli studi di Napoli "Federico II", 2009.

[9] J. M. Pollard, *Monte Carlo methods for index computation mod p*, Mathematics of Computation, 32 (1978), pp. 918–924.

[10] J. H. Silverman, *The arithmetic of elliptic curves*, vol. 106 of Graduate Texts in Mathematics, Springer, 1986.

[11] E. Teske, *Speeding up Pollard's rho method for computing discrete logarithms*, Lecture Notes in Computer Science, 1423 (1998), pp. 541–554.

[12] ———, *On random walks for Pollard's rho method*, Math. Comput., 70 (2001), pp. 809–825.

[13] P. C. Van Oorschot and M. J. Wiener, *Parallel collision search with cryptanalytic applications*, Journal of Cryptology, 12 (1999), pp. 1–28.

[14] H. C. A. van Tilborg, *Encyclopedia of cryptography and security*, Springer-Verlag, 2005.

Results and Programming Techniques from the CR3x+1 Project

Giuseppe Scollo[1]
[1]*University of Catania, Department of Mathematics and Computer Science*
Catania, Italy, scollo@dmi.unict.it

***Abstract*—The CR3x+1 project runs a parallel algorithm on the COMETA Grid Infrastructure for the search of Class Records in the 3x+1 Problem. The search up to 2^{59} was recently completed, following a 15-month computational effort that delivered 149 new Class Records. Besides presenting these results, a few techniques adopted in this project are introduced that are aimed at keeping the search proceed on schedule despite of occasional failures and delays of running jobs. The ideas behind these techniques seem easily adaptable to a wider class of search programs over a countable search space. Examples of problems solved by this kind of programs are provided, and an attempt is made to characterize such a wider applicability class. A summary of the measured performance of the CR3x+1 search program is provided, along its evolution throughout several running versions. An outline of a future research direction concludes the paper, relating to the aim of full automation of job control and management for this application, and in perspective for any suitable program in the aforementioned wider class.**

***Index Terms*—COMETA, Collatz Problem, Grid programming, recursive parallelization, job control and management.**

I. Introduction

FOLLOWING the results published in [7], this paper reports on the progress made in the COMETA Grid Infrastructure by the CR3x+1 project, since Spring 2008 up to Winter 2009. This project owes its name to the target of its running application, which implements a parallel algorithm for the search of Class Records in the 3x+1 Problem [1], also known as Collatz Problem and under several other names.

To the benefit of the unfamiliar reader, the basic terms of the search target are recalled below, whereas a detailed account about the 3x+1 Problem, which is *not* in the focus of the search target, is provided by the paper just cited, and plenty of additional references may be found in the comprehensive annotated bibliography on this problem and its several extensions and generalizations, that is maintained by the same author on *arXiv* [2],[3].

A. 3x+1 Class Records, and the like

Subject of interest here is the behavior of the iterates of the function which takes odd integers x to $3x+1$ and even integers x to $\frac{x}{2}$. The $3x+1$ Conjecture asserts that, starting from any positive integer x, repeated iteration of this function eventually produces the value 1; or, in other words, that it has the $1 \to 4 \to 2 \to 1$ cycle as (unique) attractor. One may classify the eventually periodic $3x+1$ trajectories ending up in this cycle by their finite transient length, also referred to as the *delay* of a trajectory, and aims at finding minimal elements of delay classes, that are referred to as *class records*. More precise definitions of these and related concepts are as follows. Let f denote the subject function, and f^n its n-fold iterate, then the following definitions refer to the (discrete) dynamics of iterated f:

- *trajectory* at (origin) x: the infinite sequence $x{=}f^0x,\ f^1x,\ f^2x,\ \ldots$;
- *delay* of (trajectory at) x: the smallest n such that $f^nx{=}1$;

- *delay class* d: the set of those x which have delay d;
- *class record* (CR): the smallest member of a delay class;
- *delay record* (DR): an x such that every $y < x$ has a lower delay.

A couple of remarks are in place:

- every delay class is populated (2^d is the largest member of delay class d), so a CR exists in every delay class;
- every DR is a CR (by the definitions of these concepts), but the converse does not hold (for example, 5 is the CR of delay class 5, but it is not a DR, since 3 has delay 7; the latter happens to be a DR).

The computational challenge posed by the CR search is not just the trivial consequence of the fact that it will never end, since not only the search space is infinite but also the search target, that is the set of CR's, is infinite. It rather consists of a twofold aspect of the CR distribution:

1) exponential decay of their average density;
2) logarithmic growth of the average delay of trajectories.

The design of an efficient, parallel algorithm for CR search was addressed at the very outset of the CR3x+1 project. The interested reader is referred to our cited previous work for an overview and technical details of the algorithm optimization techniques adopted in this project.

B. 3x+1 CR search efforts

A distributed 3x+1 CR search effort has been set up since quite a few years by [5], who maintains a website with results and status of the ongoing progress of that endeavour. This has produced 2016 Class Records so far, exploring the search space up to $60200 \cdot 10^{12}$. The present CR3x+1 search on the COMETA Grid Infrastructure arose as a follow-up of that effort, and was primarily aimed at expanding the rate of search space exploration thanks to the computational power here available, supplemented by the aforementioned optimization techniques. Indeed, a tenfold widening of the explored search space is the primary outcome of this endeavour to date.

II. CR3X+1 PROGRESS AND RESULTS

15-month CR search in the COMETA Grid Infrastructure (9/2007 – 12/2008) delivered 149 new Class Records, by exploring the search space up to 2^{59}. To date this has been raised up to $6 \cdot 10^{17}$ and 7 new CR's have been found. The list of 51 CR's found by this search until March 2008 is available in [7]; those found afterwards, up to date, are listed here, in the Appendix.

Besides the aforementioned results, the progress made by the CR3x+1 project includes major software enhancements, to cope with a software upgrade of the job management system and command-line user interface operating in the COMETA Grid Infrastructure. Those enhancements do not affect the search algorithm implementation, where only a minor change in the output format turned out to be convenient. A major scripting effort proved rather necessary to exploit new functionalities made available by the Grid software upgrade, as well as to take new resource allocation constraints into account. The latter were introduced, along with the Grid software upgrade, by new Grid management policies aimed at enforcing fair use of computational resources. The following details are needed, in order to appreciate the technical evolution of the CR3x+1 application that is reported in later sections below.

In Summer 2008, the COMETA Grid middleware installation was upgraded to *gLite 3.1*, including an upgrade of the Workload Management System to the most recent version, hereafter referred to as *WMS*. In order to enable a smooth transition to the new user interface and functionalities provided by the upgrade, the new *WMS* was actually adjoined to the legacy one, hereafter referred to as *EDG* (acronym of the European DataGrid project), but this is going to be disposed of in the near future.

A convenient, new *WMS* functionality is the support of *collection*-type jobs, that is, the ability to submit a bunch of parallel jobs, termed

nodes of the collection, as if they were a single one, the collection *parent* job. This greatly reduces the submission waiting time, and it also simplifies the inquiry of the status of all jobs in the collection. This functionality proves very convenient to implement, by appropriate scripts, the recursive parallelization technique described in Section IV below. However, in order to overcome the unavailability of resources following from temporary *WMS* configuration problems, an implementation of recursive parallelization has been also developed to run under *EDG*.

The aforementioned management policies [4] were introduced in the COMETA Grid Infrastructure near the end of Summer 2008. These constrain the number of jobs that may be concurrently running per user and per site, depending on job CPU time requirements. For ordinary applications, three types of job queues are available at each site. Let T denote the upper bound on CPU time, N the upper bound on the number of concurrent running jobs per user; the three queue types are as follows:

- *short*: T = 15 min., N = 64;
- *long*: T = 12 hours, N = 32;
- *infinite*: T = 21 days, N = 16;

Before the introduction of the N upper bounds, the CR3x+1 jobs used to be ordinarily submitted to the *infinite* queues only, with CPU time in the 4–6 day range, whereas use of the other queues was limited to the recovery purposes which are described in Section IV below. Under the new policy, both *infinite* and *long* queues are ordinarily made use of, whereas use of the *short* queues is confined to running recovery job collections.

III. Uncertainty and partial failures

Practical problems with Grid computing of our concern mainly fall into three categories:

1) uncertainty about availability of resources (CPU's in the CR3x+1 application case),
2) occasional failures/delays of running jobs,
3) unreliability of job status information.

Two problems surface in the first category: 1) the fact that the number of Free CPU's reported by the Grid Information Service (BDII) is not always faithful, hence CPU availability may only be confirmed by the successful outcome of actual submission attempt, and 2) uncertainty about persistence of availability of an allocated CPU, for the whole duration of job execution; this problem is obviously significant mainly for jobs requiring several days of CPU time.

Problems in the second category are mostly due to hardware or system failures at specific Worker Nodes (WN) in the Grid computing elements (CE) hosting them.

If the extent of problems of either category is fairly limited, the job control and recovery techniques described below prove effective. Problems in the third category present the most challenging difficulties; they give rise to the core of a future research direction, as it is mentioned in Section VII.

IV. Recursive parallelization

Job control and parallelization prove effective to cope with the first two aforementioned problem kinds, for:

small is easier (to get Done (with Success) ;)

that is: if a job fails, rather than restarting it, reassign its computational task to a collection of parallel jobs; if these are small enough, then many more CPUs are available in the COMETA Grid, under the policies mentioned in Section II. Furthermore, short jobs have much higher chances to succeed anyway.

The basic CR search algorithm deployed on the COMETA Grid is easily amenable to a simple kind of parallelization, based on search space partitioning, where independent, noncommunicating processes explore separate intervals. They produce *CR candidates*, while a *merge* process combines their outcomes together with previously found CR's, as depicted in Figure 1.

Each parallel process explores a given interval I_k of the search space, referred to as its search *slice*, and yields a set of *CR candidates* (CRC) in that slice, that consists of the lowest elements in delay classes that happen to be populated in the explored slice. Thus, if all

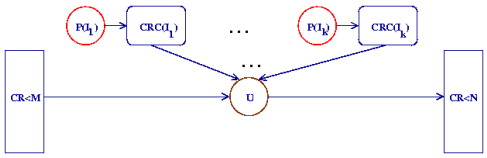


Fig. 1. Simple structure of CR parallel search

CR's below a given integer M are known, the new CR's above M and below $N > M$ can be determined by partitioning this search interval into any convenient number of slices, then running the parallel search processes on them, and finally selecting the best (*i.e.* lowest) CRC found for each delay class that is not populated by any number below M.

It is apparent that the algorithm depends neither on the size of the search interval nor on the slice size, which are just parameters of its execution. This fact may be conveniently exploited to effectively manage the fairly frequent case that some of the running processes fail to complete their search. When a process failure is detected at a time when most of the other processes have completed their search, the Grid reallocation mechanism would entail doubling the total waiting time for the parallel search completion. This can be avoided by conveniently *subslicing* the missing slice and launching a parallel search on it, thus by straightforward exploitation of a space-time tradeoff.

Recursive parallelization simply is the iterated application of this mechanism to missing search intervals because of job failure or excessive delay. A search taking a few days by a single job will take a few hours by a collection of, say, 32 or even 16 parallel jobs, and if some of these fail, by recursive parallelization the missing tiny bits of the planned recovery will only take a few minutes to be filled up.

A. Parallel subslicing

The first implementation of recursive parallelization for the CR3x+1 application exploited the representation of the search space partitioning in the application parameter structure. The chosen representation obeys to an ergonomic principle, that is: keep typical parameter values small, despite of the fairly large size of numbers at stake. This rule should help one to avoid, as well as to more easily detect, typing errors in the provision of actual parameter values to application instances (to be) submitted for execution. The rule is easily enforced by using (integer) binary logarithms for actual parameter specification. An instance of the CR3x+1 search program is thus parameterized as follows:

$$\textit{findCR} \quad a \quad s \quad c \quad b \quad z$$

with the following terminology and meaning of the displayed parameters:

- a: search *basespace*
- s: search *subspace*
- c: search *slice*
- b: search *subspace size*
- z: search *slice size*

which specifies the actual search subspace [M,N], with $M = 2^a + s \cdot 2^b$, $N = 2^a + (s+1) \cdot 2^b - 1$, and therein the $(c+1)$-th 2^z-size search slice, that is the interval $[M + c \cdot 2^z,\ M + (c+1) \cdot 2^z - 1]$, under the following, fairly obvious constraints on parameter values: $a > 0$, $0 < b \leq a$, $0 < z \leq b$, $0 \leq s < 2^{a-b}$, $0 \leq c < 2^{b-z}$.

Subspace [M,N] is thus searched by running the corresponding 2^{b-z} instances of *findCR* which only differ in their c slice values.

Now, suppose the job running the k slice instance fails. Then one may parallelize the search on the missing slice by viewing the slice interval as a subspace, and thereby *subslicing* the slice. The appropriate parameters of the parallelized subslicing search

$$\textit{findCR} \quad a' \quad s' \quad c' \quad b' \quad z'$$

are determined by a straightforward linear transformation of those of the parent search, for the given k and for a chosen subslice size $z' < z$, with c' ranging through the whole allowed range: $a' = a$, $s' = s \cdot 2^{b-z} + k$, $b' = z$.

The CPU time bounds put on the three queue types mentioned in Section II, together with measured execution times of the *findCR* program made it convenient to adopt a 2-level

subslicing scheme, with the following size parameter values for the different queue types:

- *infinite*: $b = 50,\ \ z = 44$;
- *long*: $b' = 44,\ \ z' = 40$;
- *short*: $b'' = 40,\ \ z'' = 34$.

B. Sequential subslicing

If CR's were the only interesting findings of the CR3x+1 application, then there would be little reason to definitely favour smaller slice sizes to larger ones. The higher success chance enjoyed by short jobs has tradeoffs in the relatively higher turnaround time (it usually takes 20 minutes to get back the outcome of a 10-minute CPU time job), and in the larger number of output files generated thereby. However, the parallel processes displayed in Figure 1 produce CR *candidates*, and while most of them turn out not to be CR's after the merge process execution, they are all potentially interesting findings, by the following fact.

The curious reader who has had already a first go through the CR tables in the Appendix may have wondered what's behind the rarity of "surprise!" occurrences in the righmost column. In other words, the puzzling question is: How does it happen that most of the CR's found by this search were already known to be good CR candidates, for the respective delay classes? The answer is found in the existence of *approximate* search algorithms, which nonexhaustively explore selected portions of the search space for such candidates, using knowledge of good CR candidates—not necessarily CR's, though. The outcomes of such an algorithm were communicated to the author by Eric Roosendaal [6], and the rarity of surprises in the Appendix testifies to the strength of the findings by that algorithm. Conversely, the CR *candidate* outcomes of the CR3x+1 search in the COMETA Grid may be given as new inputs to Roosendaal's algorithm, which may thus find new, stronger candidates.

Now, since not only CR's but also CR candidates are an interesting outcome of the CR3x+1 search, the desirability of a uniform subslicing scheme comes to play. For, a CR candidate produced by a slice search process is such if it is the best candidate for its delay class *in the slice searched*. So, the wider the slice size, the smaller the number of CR candidates found by the CR3x+1 application. Furthermore, the move to ordinary use of *long* queues, in addition to *infinite* ones, under the new resource allocation policy mentioned in Section II above, calls for adoption of the aforementioned 2-level subslicing scheme *regardless of parallelization*, if uniform selection of CR candidates is desired—that is to say, all of them ought to be best candidates in slices of the same size.

The implementation of this requirement in the CR3x+1 application adjoins *sequential* subslicing to (recursive) parallel subslicing, simply by stipulating that 1) all *findCR* executions have the same values of size parameters, viz. those of the lowest level, yet 2) each script run by a job submitted to a *long* or *infinite* queue, loops through a finite sequence of such executions, so as to fully search the relevant, higher-level (sub)slice. A second implementation of parallel subslicing is also required, for, according to the first clause, no parameter transformation is applicable to size parameters of *findCR* calls, but only to the "address" parameters s and c, for a given basespace a. As a matter of fact, the linear transformation specified in the previous subsection now applies to parameters of the scripts where those calls occur, and it is implemented inside those scripts in the sequential subslicing case.

V. Widening the applicability class

It is apparent that the parallelization and subslicing techniques exposed above do not depend on any particular feature of the CR search algorithm, nor of its implementation in the COMETA Grid Infrastructure, but only on (the chosen representation of) the search space partitioning. One is thus presented with an interesting *reusability problem*: to abstract from the specific (3x+1) Problem at hand, to make the aforementioned Grid programming techniques applicable to a wider problem class. Here are a few examples of problems in this class:

- search of unknown prime numbers;

- search of unknown twin prime pairs;
- CR search for classes of (consecutive) prime gaps (this problem is connected to the Goldbach conjecture).

An informal characterization of the applicability class prescribes:

1) applicability of the parallel search structure depicted in Figure 1, where the CR and CRC outputs are meant to be those of the problem at hand, and
2) hierarchical partitioning of the search space, with address and size parameters of its search intervals like those specified in the CR3x+1 application case.

The latter prescription does not require the logarithmic base of the representation to be 2 (any higher integer may do as well, 2 just happens to be the smallest suitable one), nor the depth of the subslicing hierarchy to be 2—this is just convenient in the CR3x+1 application case because of the queue management policy in the COMETA Grid Infrastructure, together with the distribution of CPU time performance exhibited by CR3x+1 search processes. A toplevel overview of this distribution is presented next.

VI. CR3x+1 SEARCH PERFORMANCE

Early performance measurements for the CR3x+1 application are reported in [7], giving average CPU time and standard deviation statistics of CR search in basespaces 55 and 56, thus for numbers below 2^{57}. Here we present similar, follow-up data for basespaces 57 and 58, which were fully covered, and for the explored portion of basespace 59 to date, that comprises its first 24 (out of 512) 2^{50}-size subspaces. Table I summarizes the aforementioned statistics, by similar conventions as in Table II of the cited paper, that we recall here for the sake of self-containedness.

Each row refers to a group of search subspaces processed by the same program version and with the same optimization parameter values. The program version is displayed in the first column. The next two columns tell which search intervals do the statistics refer to, by giving the basespace in the second column and the explored interval of its 2^{50}-size subspaces in the third column. The average CPU time and its standard deviation are measured in seconds and computed over the 64 2^{44}-size toplevel slices of each subspace; their ranges are reported in the fourth and fifth column, respectively.

TABLE I
CPU TIME PERFORMANCE STATISTICS

v	*a* *s*	average	s.d.
6.03	57 0–9	[343189–365047]	[26793–41979]
6.04	57 10–33	[337754–382158]	[24252–54268]
6.05	57 34–43	[348373–365971]	[31443–42787]
6.06	57 44–78	[338352–415004]	[24788–117136]
6.07	57 79–127	[333970–391010]	[24300–73351]
6.08	58 0–18	[362263–384922]	[26441–52488]
6.09	58 19–68	[358482–402003]	[27861–68733]
6.10	58 69–69	[368772–368772]	[29518–29518]
6.11	58 70–138	[337881–407845]	[23206–50585]
6.12	58 139–194	[360797–422676]	[24684–81774]
6.13	58 195–198	[398003–413196]	[24648–45265]
6.14	58 199–255	[384373–423242]	[8476–45754]
7.01	59 0–0	[423735–423735]	[13246–13246]
7.02	59 1–23	[396300–421982]	[8134–40764]

VII. CONCLUSION AND FUTURE RESEARCH

The outcomes of a 17-month computational effort have been reported, whereby a tenfold widening of the previously explored search space has been achieved and the list of known 3x+1 Class Records has been expanded by 156 new entries. The core software of the CR3x+1 application, which has delivered these results by implementing an optimized parallel search algorithm on the COMETA Grid Infrastructure, has reached maturity and has proven highly stable (it will be made freely available as soon as its documentation will be completed). On the contrary, the job control and management scripts whereby care is taken of the actual deployment of the application over the highly dynamic configuration of the Grid, has undergone unrelating evolution, primarily to cope with changes both in the Grid management

software and in the resource allocation policies thereby adopted.

Further evolution of job control and management is a future research direction, aimed at full automation of these tasks for the CR3x+1 application, and in perspective for any suitable program in the wider class characterized in Section V, in spite of the problems mentioned in Section III—especially the third kind thereof.

APPENDIX

Tables II–IV list the 105 Class Records found after those published in [7], thus starting from March 2008 until the end of February 2009, together with their discovery dates. Commas are inserted in the CR values in these tables, to improve readability. Notes in the rightmost column apply to a few of these CR's: two of them also are Delay Records (marked by "DR"), whereas the three CR's marked by "surprise!" are those which turned out to be lower than the best candidate known, for that delay class, until the CR discovery.

ACKNOWLEDGMENT

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Programma Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] J. Lagarias, The 3x+1 problem and its generalizations, *Amer. Math. Monthly* **92** (1985) 3–23, http://www.cecm.sfu.ca/organics/papers/lagarias.

[2] J. Lagarias, The 3x+1 Problem: An annotated bibliography (1963–2000), preprint on *arXiv*, v. 9, http://arxiv.org/abs/math/0309224v9.

[3] J. Lagarias, The 3x+1 Problem: An Annotated Bibliography, II (2001-), preprint on *arXiv*, v. 2, http://arxiv.org/abs/math/0608208v2.

[4] A. Falzone, *Known limitations on PI2S2 infrastructure*, Consorzio COMETA, PI2S2 Project, Catania (Italy), 22 September 2008, https://grid.ct.infn.it/twiki/bin/view/PI2S2/KnownLimitationsonPI2S2Infrastructure.

[5] E. Roosendaal, *On the 3x+1 problem*, http://www.ericr.nl/wondrous.

[6] E. Roosendaal, personal communication.

[7] G. Scollo, Looking for Class Records in the 3x+1 Problem by means of the COMETA Grid Infrastructure, in: R. Barbera (Ed.), *Proc. Symp. "Grid Open Days at the University of Palermo"*, Palermo (Italy), 6–7 Dec. 2007, Consorzio COMETA, Catania (2008), pp. 255–263.

[8] A. Sortino, *Metodi di controllo dei job nelle griglie computazionali*, Graduation Thesis, University of Catania, CdL Informatica Applicata (2008).

TABLE II
CLASS RECORDS BY THE COMETA SEARCH

date	*delay*	*CR*	*note*
07/03/2008	2027	145456,961990,848420	
07/03/2008	2102	148413,243647,555247	
07/03/2008	2071	151176,074945,747355	
07/03/2008	2009	152252,737489,082415	surprise!
10/03/2008	2053	154716,354113,071847	
10/03/2008	2252	157348,943716,351847	
10/03/2008	2084	157901,726856,603887	
14/03/2008	2159	162714,511297,434783	
15/03/2008	2097	166349,144342,759649	
15/03/2008	2066	166685,766959,953535	
23/03/2008	2079	178296,989423,512571	
23/03/2008	2017	181276,570919,731903	
23/03/2008	2048	181616,130563,790600	
28/03/2008	2110	184220,042842,339239	
30/03/2008	2092	187142,787385,604604	
31/03/2008	2136	190259,568858,878791	surprise!
31/03/2008	2061	190398,201108,018553	
31/03/2008	2030	193620,340831,833371	
03/04/2008	2105	197154,541443,270695	
05/04/2008	2074	200584,113101,451643	
06/04/2008	2118	203172,693893,734335	
06/04/2008	2056	206288,472150,762462	
12/04/2008	2255	209798,591621,802462	DR#132
12/04/2008	2087	210535,635808,805182	
16/04/2008	2162	218142,178682,751135	
16/04/2008	2100	221798,859123,679531	
18/04/2008	2069	222247,689279,938046	
25/04/2008	2051	232074,531169,607771	
26/04/2008	2250	236023,415574,527771	
26/04/2008	2082	236852,590284,905831	
26/04/2008	2126	240797,266837,018471	
29/04/2008	2157	244071,766946,152175	
01/05/2008	2095	249523,716514,139473	
01/05/2008	2064	250028,650439,930303	
09/05/2008	2139	253679,425145,171721	
13/05/2008	2108	262872,721924,360926	
13/05/2008	2245	265526,342521,343743	
13/05/2008	2077	267445,484135,268857	
18/05/2008	2059	275051,296201,016617	
18/05/2008	2152	275061,764347,852959	
19/05/2008	2214	277152,210816,179199	
19/05/2008	2121	277960,972098,942063	
19/05/2008	2258	279731,455495,736617	DR#133
20/05/2008	2134	285389,353288,318187	

TABLE III
CLASS RECORDS BY THE COMETA SEARCH (CONT'D)

date	*delay*	*CR*	*note*
23/05/2008	2103	295731,812164,906041	
23/05/2008	2072	296330,252373,250729	
26/05/2008	2116	304759,040840,601503	
31/05/2008	2147	308903,330041,223847	
31/05/2008	2054	309432,708226,143694	
31/05/2008	2253	314697,887432,703694	
06/06/2008	2129	321063,022449,357961	
11/06/2008	2160	325429,022594,869566	
11/06/2008	2098	332698,288685,519297	
11/06/2008	2067	333371,533919,907070	
22/06/2008	2173	342838,805696,652711	
25/06/2008	2111	350496,962565,814567	
29/06/2008	2248	354035,123361,791657	
29/06/2008	2080	355278,885427,358747	
30/06/2008	2093	358949,884773,873129	
30/06/2008	2124	361195,900255,527707	
05/07/2008	2155	366107,650419,228263	
05/07/2008	2062	366735,061601,355489	
10/07/2008	2261	372975,273994,315489	DR#134
18/07/2008	2137	380519,137717,757582	
24/07/2008	2106	394309,082886,541390	
24/07/2008	2075	395107,003164,334305	
24/07/2008	2243	398289,513782,015615	
30/07/2008	2119	406345,387787,468670	
30/07/2008	2181	407039,301558,612711	
03/08/2008	2150	412592,646521,779439	
03/08/2008	2212	415728,316224,268799	
04/08/2008	2256	419597,183243,604924	
11/08/2008	2132	428084,029932,477281	
12/08/2008	2163	433905,363459,826089	
12/08/2008	2101	433924,493696,872063	
17/08/2008	2070	444495,378559,876092	
23/08/2008	2114	457138,561260,902255	
25/08/2008	2145	463354,995061,835771	
30/08/2008	2251	472046,831149,055542	
02/09/2008	2127	481594,533674,036942	
03/09/2008	2189	484213,647044,074527	
07/09/2008	2158	488143,533892,304350	
07/09/2008	2096	489496,481087,649823	
07/09/2008	2220	492715,041450,985243	
07/09/2008	2264	497300,365325,753985	DR#135
16/09/2008	2171	502013,107133,492583	surprise!
17/09/2008	2140	507358,850290,343442	
01/10/2008	2202	509154,959181,631323	
19/10/2008	2078	522719,142645,844807	

TABLE IV
CLASS RECORDS BY THE COMETA SEARCH (CONT'D)

date	*delay*	*CR*	*note*
24/10/2008	2109	525745,443848,721851	
29/10/2008	2246	531052,685042,687486	
24/11/2008	2277	539483,373894,066267	DR#136
29/11/2008	2122	541793,850383,291561	
07/12/2008	2153	549161,475628,842395	
14/12/2008	2215	554304,421632,358398	
22/12/2008	2259	559462,910991,473233	
31/12/2008	2135	570778,706576,636374	
31/12/2008	2197	571753,422216,761887	
05/01/2009	2073	576770,710941,544863	
06/01/2009	2166	578540,484613,101451	
06/01/2009	2104	578565,991595,829417	
12/01/2009	2228	583958,567645,612139	
26/01/2009	2272	589393,025571,263983	
03/02/2009	2241	597434,270673,023423	
10/02/2009	2148	601314,192936,703339	

A New Metric to Describe the Efficiency of Data Communication Networks

Elena Marchei[1], Sandro Meloni[2], Saverio Di Blasi[1],Vittorio Rosato[3,4]

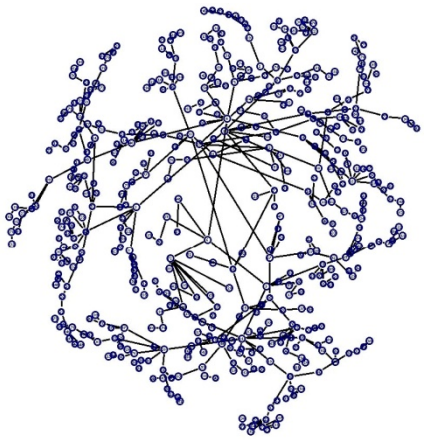
[1] *Ente per le Nuove Tecnologie, l'Energia e l'Ambiente (ENEA), Portici Research Center, Portici (Italy)*
[2] *Dip.to Ingegneria Informatica, Università di Roma Tre, Via della Vasca Navale, Roma (Italy)*
[3] *Ente per le Nuove Tecnologie, l'Energia e l'Ambiente (ENEA), Casaccia Research Center, Roma (Italy) and*
[4] *Ylichron Srl, Via Anguillarese 301, 00123 Roma (Italy)*
elena.marchei@enea.it, saverio.diblasi@enea.it, sandro.meloni@gmail.com, rosato@enea.it

Abstract— **This work has been focussed on the characterization of the congestion transition, occurring on model networks when traffic volume increases above a given threshold, and on its dependence on the network's topology. We have attempted to describe the transition using a number of quantities which might be measured on the different network's components. We have evaluated several quantities and showed how they behave above and below the congestion threshold. In particular we were interested in defining quantities which could provide each network's node a certain "awareness" of the global state of the network. Synthetic Internet-like and Random networks have been generated by using well-established growth mechanisms; data traffic behavior on the networks has been simulated by using either a simple traffic model (hereafter referred to as Traffic-Model) which reproduces the basic features of the traffic protocols and the NS2 tool which allows to simulate the whole TCP/IP stack. Data traffic undergoes a phase transition from a low-traffic phase of normal behaviour to a congested phase, characterized by a rapid increase on the value of the average delivery time of data packets and a sharp reduction of the Quality of Service, which accounts also for the number of effectively delivered data packets. We propose the use of a new metric to individuate the onset of the congestion transition. The proposed quantity, referred to as "node's temperature" allows to integrate, in an unique value, information on the local and the global state of data traffic. Node temperature can also be used to implement new adaptive routing strategies able to prevent (or delay) the congestion transition.**

NETWORK TOPOLOGIES

Two different network's topologies (both formed by 1000 nodes and bi-directional links of equal bandwidth) have been generated; a random network, R-net, by using the Erdos-Renyi growth mechanism [1] and a Internet-like network, I-net, grown on a mechanism which couples Preferential Attachment [2] to Triad Formation [3], as proposed in [4].

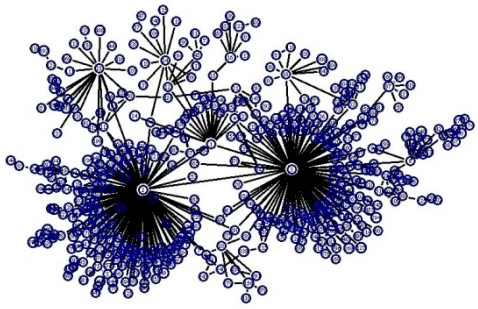
Example of a R-net



The Erdos-Renyi model is a random graph generation model that produces graphs where the sticking probability of new nodes, in the growth mechanism, is independent on the node where the sticking is produced. In the resulting structure, nodes have approximately a similar

degree and the degree distribution is a poissonian distribution.

Example of an I-net



The Internet network has been established to display a topology which is far from being random. It belongs to the class of "scale-free" networks, where nodes with higher degree have stronger ability to grab links added to the network. In the growth process, nodes with higher degree have a larger probability to link with new nodes. This mechanisms produces a probability distribution of nodes $P(k) \sim k^{-\gamma}$ where γ is a constant whose value is typically in the range $2<\gamma<3$. To better mimic the Internet topology, one must add to this mechanism a mechanisms called "triad formation" which allows to generate scale-free networks with a high *clustering* coefficient

TRAFFIC-MODEL

Each node of the network represents a router and it has a buffer of a limited size (in terms of number of packets which it can contain) able to host data packets waiting to be delivered; to a first approximation, buffers have equal size for all the nodes of the network.
With a specific frequency, nodes send a small data packet which flows along the network via the shortest path (a fixed-path routing is primarily established to rule the packets path). They are hosted in the node's buffer, if buffer contains previously arrived packets which must be handled with priority, or dispatched toward to next node if the buffer nodes is empty. Buffers, in fact, handle data packets with a FIFO (First-In-First-Out) policy. If a data packet arrives at a node whose buffer is full, it is dropped and, in a first approximation, it is lost.
Simulations of data traffic have been performed by selecting a value λ measuring the traffic intensity: λ represents the frequency of emission of a new data packet by each node. $\lambda = 0.1$ will thus indicate that each node has a probability of 10% to generating, at each simulation time-step, a new data packet, directed towards a generic j node.

NETWORK SIMULATOR NS2 AND ENEA-GRID

NS2 is a discrete event simulator that together with its companion Nam, the Network Animator, form a very powerful set of tools for study of the real networks [www.isi.edu\nsnam\ns].
NS2 allows to study e.g. wired and wireless (local and satellite) networks; create various routing algorithms; traffic sources like web, ftp, telnet, cbr, stochastic traffic; failures, including deterministic, probabilistic loss, link failure; various queuing disciplines.
When used in NS2, both R-net and I-net models have been designed with a propagation delay of 30ms (which could be representative of the times of signal processing) and queues type Drop-Tail, whose policy is to serve packets arriving with criterion FIFO (First In First Out).
Each node of the network is, at the same time, both source and destination.
The couples source-destination time are generated in random way. We have used, as transport protocol, an agent connectionless UDP (User Datagram Protocol) and as application traffic, a source with a rate constant CBR (Constant Bit Rate). Simulations of data traffic have been performed by selecting a value of the source rate λ as *packets per second*.
To evaluate the network efficiency, we have used a Unix script that allows us to launch several simulations on different Linux hosts of ENEA-GRID. Each job runs simultaneously in batch mode. The input parameters, used to

automatically vary source rate, change for each simulation. The output files are stored in a unique AFS (Andrew File System) volume and are visible from all hosts on which the distributed geographic file system is configured.
In the custom traffic simulator, described in [4] we have designed a simple point-to-point connection mechanisms where packets are routed by a deterministic or a probabilistic routing protocol connecting shortest paths, still with a buffer FIFO policy.

ANALISYS OF NETWORK'S EFFICIENCY

Simulations have been performed by starting from empty networks. The results of Traffic-Model are then compared with the simulation results coming from the tool NS2.
We have focussed our attention on different quantities related to the network function and efficiency. The efficiency of the network to sustain a given traffic will be estimated through the evaluation of the average delivery time <***T***> of data packets which is defined as follows:

$$< T >= \frac{1}{N_d} \sum_{i}^{N_d} t_i$$

where t_i is the delivery time of the effectively delivered i-th packet; N_d data packets are delivered out of the N_s packets effectively sent (the difference among these numbers is the total number of packets dropped because the limited buffer sizes).
The *Quality of Service* (*QoS*), which accounts for both the delivery time and the number of packets which are dropped, can be defined as follows:

$$QoS = \frac{N_d}{N_s} \frac{T_{th}}{< T >}$$

where T_{th} is the minimum time needed, by a data packet, in average, to arrive from the origin node to the destination one.

ANALISYS OF PRELIMINAR RESULTS

The first step of our work was to assess the qualitative correctness of the Traffic-Model to reproduce an effective TCP/IP data flow. We compared the results of Traffic-Model to those obtained by using NS2 tool which resulted to be qualitatively similar.
In both the Traffic-Model and the NS2 tool, the network's traffic undergoes a phase transition at certain λ values. Above a given threshold, the delivery times, both for Random network and Internet-like network (Appendix A), becomes increasingly large and, as consequence, the efficiency of the networks, the Quality of Service decreases (Appendix B).
The future step is to use a new metric to individuate the onset of congestion transition. The idea is to evaluate a further quantity, referee to as "node's temperature" which measures the effects of the traffic load that the node can sustain, let us call this quantity node's temperature $\Theta_{(t_i)}$.
At a given simulation time t_i, we could define, on each node, the following quantity:

$$\Theta_{(t_i)} = \begin{cases} \frac{n_{B(t_i)}}{l_B} \frac{< \Delta T_j >_{(t_i)}}{T_{th}} & n_{B(t_i)} \neq 0 \\ 0 & n_{B(t_i)} = 0 \end{cases}$$

where $n_{B(t_i)}$ is the current number of packets in the node's buffer; l_B is the total length of the buffer; $<\Delta T_j>(t_i)$, (j = 1, n_B) is the average "age" of the packets contained in the node's buffer, and T_{th} is the minimum time needed, in average, to arrive from the origin to the destination nodes.
The $\Theta_{(t_i)}$value is able to capture the state of node's congestion: if $\Theta_{(t_i)} \approx 0$, the node is in an "unstressed" state (buffer empty), while $\Theta_{(t_i)} > 1$ indicates a buffer almost full and the network begins to be in the congestion phase.
We can say that the simulations performed with the NS2 tool assessed the quality of the proposed model which has been, in first approximation, used to calculated the node's temperature.

A first result by simulating the Internet-like network, is shown in the figure below:

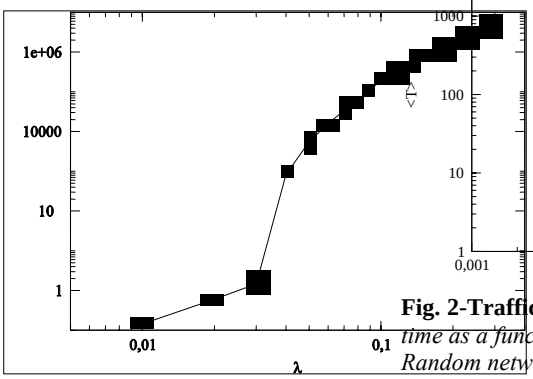


Fig. 1: *Variation of the efficiency mean temperature of the Internet-like network as a function of packet emission λ for the Traffic-Model.*

The analysis of average temperature <Θ> shows the network behavior: for 0 <<Θ> <1 the network is in the free flow state and <Θ>> 1 the network is in the congestion state. We can say that the node's temperature is a clear indicator of the network status.
The next step will be to use <Θ> to implement new adaptive routing strategies able to prevent (or delay) the congestion transition.

REFERENCES

[1] P. Erdös, A. Rènyi, Publicationes Mathematicae **6** (1959) 290
[2] A.-L. Barabàsi, R. Albert, Science, **286** (1999) 509
[3] P. Holme, B.J.Kim, Phys. Rev. E **65** (2002) 026107
[4] V. Rosato, L. Issacharoff, S. Meloni, D. Caligiore, F. Tiriticco, Physica A **387** (2008) 1689

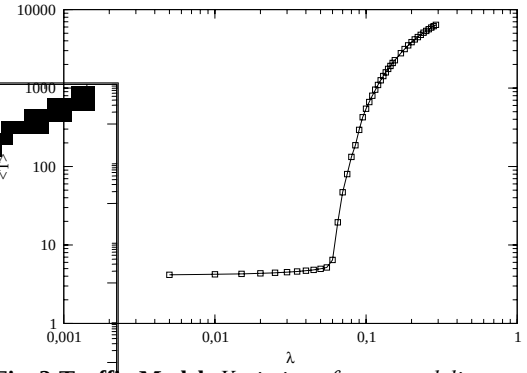


Fig. 2-Traffic-Model*: Variation of average delivery time as a function of the rate of packet emission λ for Random network. Data are in log-log scale.*

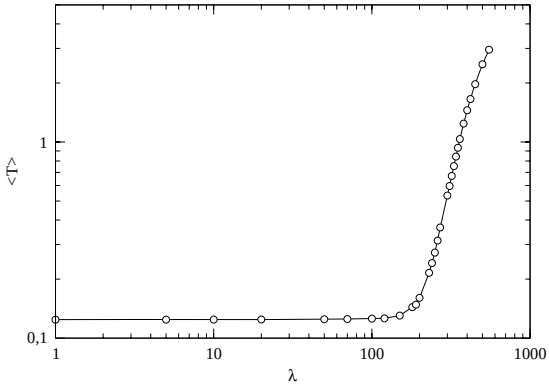


Fig. 3-NS2: *Variation of average delivery time as a function of the rate of packet emission λ for Random network. Data are in log-log scale.*

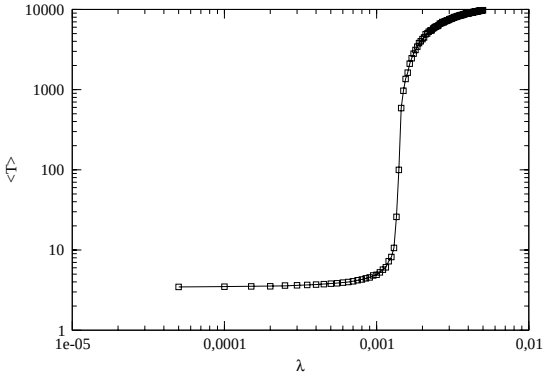


Fig. 4-Traffic-Model: *Variation of average delivery time as a function of the rate of packet emission λ for Internet-like network. Data are in log-log scale.*

APPENDIX A

AVERAGE DELIVERY TIME

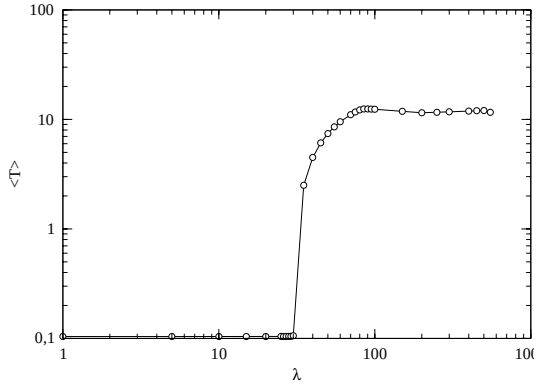


Fig. 5-NS2: *Variation of average delivery time as a function of the rate of packet emission λ for Internet-like network. Data are in log-log scale.*

APPENDIX B

QUALITY OF SERVICE

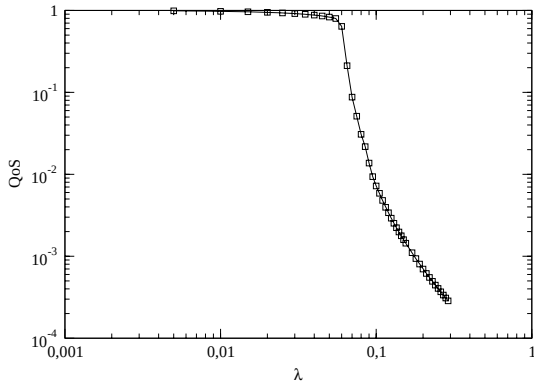


Fig. 6-Traffic-Model: *Variation of the Quality of service as a function of the rate of packet emission λ for Random network. Data are in log-log scale.*

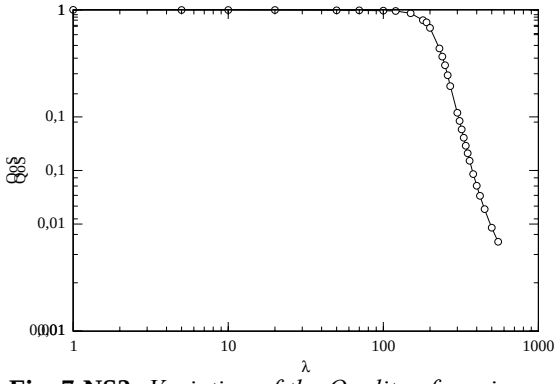


Fig. 7-NS2: *Variation of the Quality of service as a function of the rate of packet emission λ for Random network. Data are in log-log scale.*

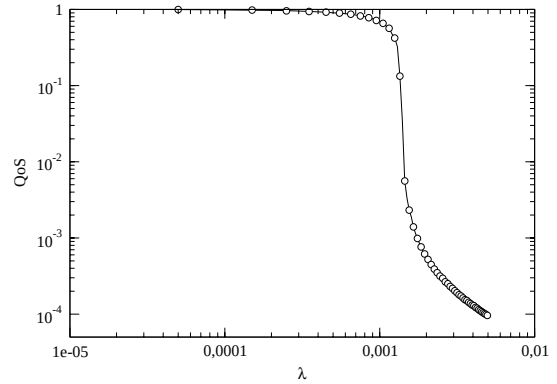


Fig. 8-Traffic-Model: *Variation of the Quality of service as a function of the rate of packet emission λ for Internet-like network. Data are in log-log scale.*

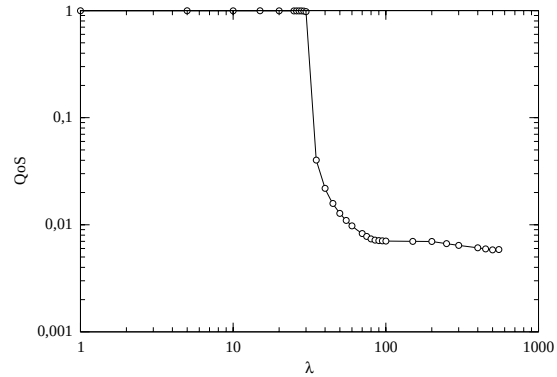


Fig. 9-NS2: *Variation of the Quality of service as a function of the rate of packet emission λ for Internet-like network. Data are in log-log scale.*

Parallel Programming in Matlab Environment on CRESCO Cluster, Interactive and Batch mode

G. Guarnieri[1], S. Migliori[2], S. Podda[3]
[1]*ENEA-FIM, Portici Research Center, Via Vecchio Macello - 80055 Portici (Naples)*
[2]*ENEA-FIM, Enea-Sede, Lungotevere Thaon di Revel n. 76, 00196 Roma*
[3]*ENEA-FIM, Frascati Research Center, Via E. Fermi 45, 00044 Frascati (Roma)*

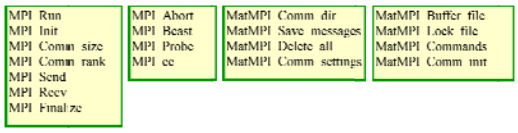
***Abstract*—Matlab (MATrix LABoratory) provides an environment for numeric elaboration and an interpreted programming language widely used in science and technology. The package is equipped with specialized libraries (toolboxes) which can solve problems in several fields. For some years the implementation of parallel programming in Matlab environment has been under study. Mathworks, the producer of Matlab, distributes toolboxes with this target, but there are also third party projects with the same aim. Among them is the "MatlabMPI" library developed by Lincoln Laboratories of the Massachusetts Institute of Technology. ENEA GRID, by the introduction of CRESCO machines, has achieved a very high computational power. So, the possibility of parallel programming in Matlab on this structure becomes a meaningful aim. We have chosen to modify and install the "MatlabMPI" library for Linux machines of ENEA GRID and particularly for CRESCO cluster. This functions library, written in pure Matlab language, implements a subset of the functions defined in the MPI standard and allows to run, at the same time, several Matlab processes both on multicore machines and computer clusters. Data exchange among the processes does not happen by real message sending and receiving, but via file system access operations. Some code changes were required to install MatlabMPI on ENEA GRID and to adapt the use of the library to CRESCO submission mode. To enable the launching of Matlab jobs on remote machines in interactive such as in batch mode we have resorted to the ENEA GRID submission tool. In the interactive mode, the use of this library along with the multithreading programming mode on multicore nodes, enabled in latest versions of Matlab, also permit to enforce a hybrid programming mode on CRESCO cluster. The user could exploit the parallelism assured by multithreading on multicore nodes and implement the communication among the nodes with MatlabMPI.**

I. INTRODUCTION

THIS paper is intended to resume our work about the parallel programming in Matlab environment in CRESCO project. It is divided into three sections. The first briefly depicts the main features of MatlabMPI library [1] that we have modified in some points and installed on the computational grid managed by ENEA. The second section shows the use of MatlabMPI on ENEA GRID, the third is devoted to the use of the library on CRESCO HPC system in batch mode. The conclusion examines the possibility of performing hybrid programming mode in Matlab environment on CRESCO cluster.

A. MatlabMPI

In recent years there have been several open source projects focused on parallel programming in Matlab environment. Many of these were set aside and actually are unsupported. The MatlabMPI library of M.I.T. is constantly updated and well supported, and there are laboratories other than M.I.T. that obtained good results using it.
The library contains the following items:



It is important to specify how parallel Matlab processes can exchange data using the library functions because file system operations are required to perform this job.
The "send" operation consists of storing the data to send in a file in a common directory that all processes can access. The name of this file contains the information about the message to exchange, the "sending" process and the "receiving" one. The "receive" operation, instead, consists of detecting the file from which to load the message by means of the information contained in its name, of waiting for the end of "send" operation and finally of loading the data with a reading operation.

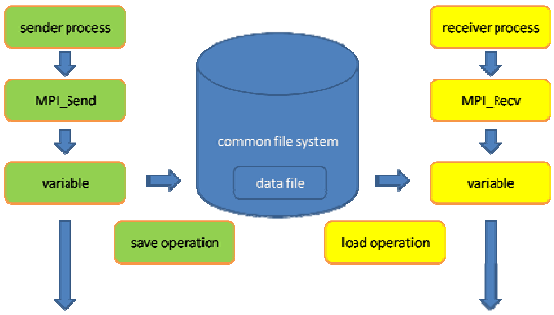


Fig.1. Graphic description of a message exchange between two parallel Matlab processes using MatlabMPI functions "MPI_Send" and "MPI_Recv".

By the use of MatlabMPI it is possible to edit Matlab scripts that can be executed in parallel on several nodes sharing the computational load among them according to the SPMD (Single Program Multiple Data) paradigm.
Since the library consists of pure Matlab code, it's possibile to use it involving heterogeneous sets of platforms as long as they support Matlab environment. On a shared memory system a single licence is enough to use all available cores, whereas on distributed memory systems the user needs a licence for every engaged node. However it is possibile to compile Matlab scripts with the mcc command, in this way only one Matlab licence is enough on distributed memory systems too.
At run time the library creates a communicator and assigns a rank at each process such as in classical MPI standard. The functions that perform the communications are the following:
MPI_Send(dest, tag, comm, data);
dest: integer that represents the rank of destination process,
tag: integer that identifies the exchange operation,
comm: communicator,
data: variable that stores the data to exchange.

data1=MPI_Recv(source, tag, comm);
source: integer that represents the rank of the source process,
tag: integer that identifies the exchange operation,
comm: communicator,
data1: variable that stores the data to exchange.

[var_1, var_2, ...] = MPI_Bcast(source, tag, comm, var_1, var_2, ...)
source: integer that represents the rank of the source process,
tag: integer that identifies the exchange operation,
comm: communicator,
var_i: variable that stores the data to exchange.

B. MatlabMPI on ENEA GRID

ENEA GRID infrastructure has been developed to provide a unified environment for all the main ENEA HPC resources. The main software components of ENEA GRID are the scheduler LSF, the OpenAFS distributed file system, the integrated Kerberos 5 authentication and a Java graphical user interface. The use of OpenAFS distributed file system, which allows users to access their home directory from every point of the grid, well supports the use of MatlabMPI because the parallel Matlab processes can simply share a common working directory. The library is presently installed on ENEA GRID and it's available for all Linux machines and for the following Matlab releases:
1) 7.0.1 24704 (R14)
2) R2007a

3) R2008a
We have modified in some points the original code to adapt the use of MatlabMPI to our computational grid. Primarily, to enable the launching of Matlab jobs on remote machines, allowing them to access the AFS file system without losing reading and writing permissions, or to be more precise to ensure token preservation for the user, we have resorted to the ENEA GRID submission tool. Moreover, in ENEA there are a limited number of Matlab licences and therefore at run time the number of available ones may be insufficient to launch all required Matlab processes, this could cause an error in the elaboration that MatlabMPI cannot detect. We have introduced a shell script that checks the number of available Matlab licences; if this number is insufficient the program exits with an appropriate error message. This script is called by MPI_Run before the stage of processes spawning. MPI_Run function was also modified in order to detect the version of Matlab that is in use and to check the number of available licences for the appropriate version. In this way the user can avoid the risk of creating pending processes with subsequent deadlock errors.
The instruction by which it is possible to launch parallel Matlab jobs in the interactive way on Linux machines of ENEA GRID has the following form:

eval(MPI_Run('script.m',n_p,machines))
where:
script.m is the Matlab code to be executed in parallel
n_p is the number of processes
machines is a Matlab cell containing the hosts on which the processors must be taken

C. MatlabMPI on CRESCO cluster

CRESCO has been an ENEA project co-funded by the MIUR (Ministry of University and Research) and had its centerpiece in the installation of a supercomputer in the ENEA Portici Research Center and in its integration in ENEA GRID.
The resulting CRESCO HPC system consists of more than 2700 cores, divided into three main sections. There are eight front-end nodes by which users can access the system and submit their jobs on computational nodes.
In order to adapt MatlabMPI library to submit jobs on CRESCO cluster in batch mode it was necessary to further modify the code. We have created some scripts to interface the library with the scheduler environment.
Users can perform parallel Matlab jobs in batch using MatlabMPI submitting the script "matMPIlauncher.sh" by the LFS command "bsub" which also allows full expression of the resource request. An example of batch submission on CRESCO cluster is the following:

bsub –q small_10m -o %J.out –e %J.err –R cresco1 –n 32 matMPIlauncher.sh

The script "matMPIlauncher.sh" reads some LSF environmental variables in which the scheduled resources for the request expressed in the previous command line are contained and launch Matlab with the M-File "matMPIscript.m". The latter passes the scheduled resources and the Matlab script to be executed in parallel as arguments to MatlabMPI functions. In batch mode the launch of the parallel Matlab jobs happens by means of "blaunch" command.

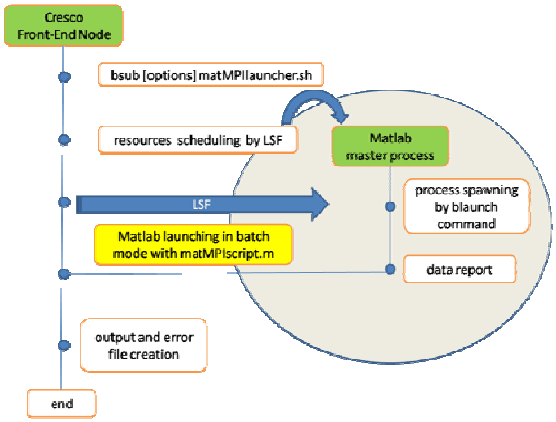


Fig.2. MatlabMPI in batch mode on CRESCO cluster. Software interface between LSF and MatlabMPI environment.

II.CONCLUSION

Now ENEA GRID and in particular CRESCO

cluster has an open source tool to distribute Matlab computations over its nodes in interactive such as in batch mode. However there are other ways to be explored to give users better optimized tools.
For example, as the CRESCO architecture is distributed with multicore nodes the best parallel programming style would be the hybrid one. This technique combines together multithreading on shared memory machines and message passing for comunication among nodes. The latest Matlab versions enable multithreading on a single multicore node thus the MatlabMPI library could fully exploit the CRESCO capabilities. In our next activities we intend to explore these way.

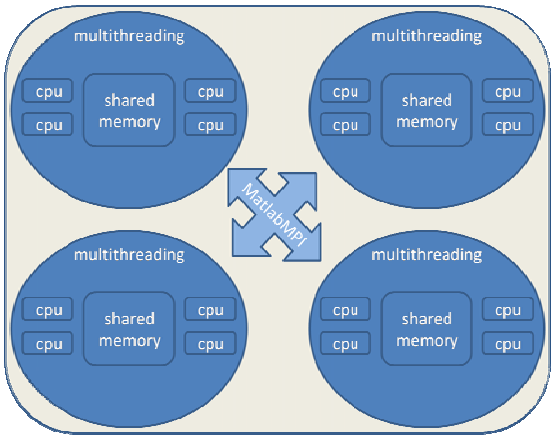


Fig.3.Hybrid programming mode on CRESCO cluster("only interactive mode").

REFERENCES

[1] ("Parallel Programming with MatlabMPI", Jeremy Kepner, Proceedings of the High Performance Embedded Computing (HPEC 2001) workshop, 25-27 Sep 2001, MIT Lincoln Laboratory, Lexington, MA).

Performance Evaluation of Parallel Algorithms on Computational Grid Environment

Simona Blandino[1,2] and Salvatore Cavalieri[1,2]

[1]*Consorzio COMETA, Catania, Italy*
[2]*Faculty of Engineering, University of Catania, Catania, Italy*
{sblandino, cavalieri}@diit.unict.it

***Abstract* — The enormous advances made in this last decade in the field of Grid Infrastructure, make very attractive their use to execute HPC parallel algorithms. As known, many factors may influence the relevant performance and scalability, among which the scheduling policy adopted at the broker level, the MPI libraries and their implementation used and the features of network adopted to connect the computing resources. Analysis of performance and scalability should be carried out for each novel Grid Infrastructure, in order to evaluate its capability to support execution of HPC parallel algorithms and to tune its configuration with the aim to improve the relevant performances, if needed. In this paper, performance evaluation of a Grid Infrastructure realised by the Consortium COMETA under the PI2S2 project, will be presented.**

***Index Terms* — Grid Computing, HPC, performance evaluation, MPI, InfiniBand.**

I. INTRODUCTION

THE increasing demand of computing resources brought about exponential growth of computers elaboration capacity.

Though the grand increasing of computing capacity, specific applications have requirements, in terms of memory, disk space and computing power, higher to get fulfilling results using a single computer. These are the mandatory demands in the all scientific fields such as Physics, Astronomy, Life Sciences, BioInformatics and Engineering.

Recently, to tackle this request, people have started using many computing resources interconnected each other within low latency communication networks.

Intuitively, we could think that the enormous computing availability spawned with computer clusters or specialized parallel computer with hundreds elaboration nodes was enough for whatever application. But the experience taught us that it is wrong.

Given the large number of applications that require huge computational resources harvesting unused CPU cycles that are potentially available for computing has become more and more important. This has fostered the spreading of systems that allow deploying programs on a large number of computers, especially when these are underutilized.

The development of this process has brought the spreading of the Grid paradigm. The Grid infrastructure allow the execution of various scientific application on a network composed of heterogeneous computer, such as parallel high-performance supercomputers, high-end servers and single workstations. The "Computational Grid" term comes from the analogy with the electric power grid. As the spread of electric power had a strong impact on the industrial and social progress of the society, similarly a simple, capillary and cheap access to computing power, memory and disk space, and whatever else having reference to computers, should push the emergence and

growth of new classes of computer applications.

II. COMETA INFRASTRUCTURE

The COMETA infrastructure is based on IBM Blade Centre H chassis each containing up to 14 IBM LS21 "blades" interconnected both with a double Gigabit Ethernet network, for normal communications with redundancy and load balancing, and a CISCO Topspin Infiniband-4X [1] network, required to provide the COMETA Grid with High Performance Computing (HPC) functionalities. The infrastructure is built with identical hardware and software at all sites. This choice was made on purpose to allow for the maximum interoperability and realizes a homogeneous environment which is a fundamental condition for an HPC Grid environment able to run distributed parallel jobs of applications adopting the MPI paradigm.

Each "blade" is equipped with 2 AMD Opteron 2218 dual-core processors with a clock rate of 2,6 GHz able to natively execute x86 32 and 64 bits binary code. Each processor has 2 GB of DDR2 RAM at 667 MHz (8 GB in total per "blade") and it is equipped with a direct communication channel to the other processor on the same motherboard. The memory controller is integrated onboard.

A. InfiniBand Communication Network & MPI Implementation

The Message Passing Interface (MPI) standard is a message passing API based on the consensus of the MPI Forum [2]. The goal of the Message Passing Interface is to establish a portable, efficient, and flexible standard for message passing that is widely used for writing message passing programs on distributed memory machines. MPI is not an IEEE or ISO standard, but has in fact, become the "industry standard" for writing message passing programs on High Performance Computing (HPC) platforms.

MPI provides many features intended to improve performance on scalable parallel computers with specialized interprocessor communication hardware. High performance can be obtained thanks to specialized vendor libraries implementing the interface. Those libraries usually come together with low latency networks such as InfiniBand or Myrinet, enabling programs to enjoy the benefits of high speed communication, and are widely used in HPC. There are two versions of the standard that are currently used: version 1.2 (shortly called MPI-1), which emphasizes message passing and has a static runtime environment (fixed size of world), and version 2.1 (MPI-2), which includes new features such as parallel I/O, dynamic process management and remote memory operations.

The notion of process and not processor is used in MPI, although commonly people talk about processor. Copies of the program are mapped to processors by the runtime environment of MPI through the agent that starts the MPI program, normally called mpirun or mpiexec. In that sense, the parallel machine can map to 1 physical processor, or N, where N is the total number of processors (or more recently *cores*) available, or something in between. Developing a message passing program requires the programmer to take care to manually distribute both operations and data among the processes. Communication routine calls between processes are inserted explicitly into the source code.

Together with the increasing of computing power some new interconnect network have appeared in the market. These networks have very low latency (< 10μsec) and wide band (multiples of Gbps). This trend allows to realize high-performance clusters that fuse the computing power of a lot number of computers and high-speed network performance. In this scenarios there are network as Myrinet and Quadrics and the newest InfiniBand (IBA) [1] architecture. Myrinet and Quadrics have been designed for high-performance computing environment, and in particular to provide an

optimized MPI implementation using in computer cluster. Because of its performance and scalability, InfiniBand has been adopted in high-performance computing (HPC) cluster environments for communication-intensive applications, including those in the fields of aerospace and defence, oil and gas exploration, fluid dynamics, weather modelling, and biology and life sciences.

As Figure 1 shows, the unified InfiniBand I/O fabric includes three basic components: host channel adapter (HCA), target channel adapter (TCA), and InfiniBand switches. HCAs are connection interfaces to CPUs; TCAs are links to storage, Fibre Channel networks, and other I/O nodes. InfiniBand switches provide the switched connections between HCAs and TCAs. InfiniBand routers can be used to connect multiple switched subnets.

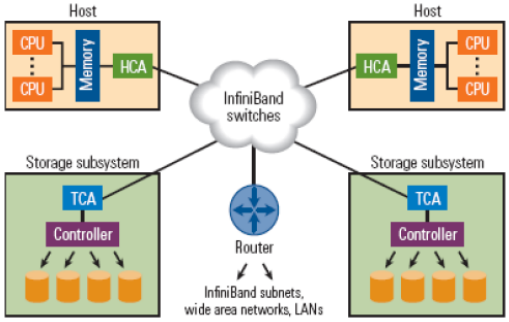


Figure 1:Architettura hardware InfiniBand

In the COMETA infrastructure there are two InfiniBand switches for BladeCenter and each blade has geared with an HCA.

The software interface for the HCA is VAPI, *Verbs-level* API [10][11]. In fact, the interface is implemented on InfiniBand *verbs* layer. The verbs interface specifies some functionalities, among which resource management at transport level, multicast, scheduling and event handling.

There are many design issues involved in putting an MPI layer on top of the Verbs Interface. MPI assumes that the underlying communication layer provides such functionality as reliability, buffer management, and flow control. MPI applications should not be involved in these issues.

Currently one of the most popular MPI implementations is MPICH [7] from Argonne National Laboratory. MPICH uses a layered approach in its design. The platform dependent part of MPICH is encapsulated by an interface called Abstract Device Interface (ADI), which allows MPICH to be ported to different communication architectures. The original MPICH implementation is MPICH1 that implements MPI-1 standard. The last implementation is MPICH2, that implements MPI-2 standard. This is an high-performance and highly portable implementation of MPI standard.

The MPI implementation on InfiniBand uses ADI2 (the second generation of Abstract Device Interface) implementation which uses VAPI as the underlying communication interface. MVAPICH and MVAPICH2 are derived libraries optimized to use the InfiniBand network. These libraries allow high performance and scalability when they are used to implement parallel applications.

III. Performance Evaluation Criteria

To test a parallel algorithm's performance, the programmer usually chooses an algorithm and evaluates its execution time on a different number of processors. In fact, parallelism is used to solve a problem in a shorter time or to solve bigger problems in the same time. S(N), the parallel algorithm's speedup on N processors, is defined as the ratio between the best execution time on a single processor and the total parallel execution time on n processor.

$$S(N) = Tser(1) / Ttot(N)$$

In 1967, Amdahl formulated a speedup law for computer systems. This law has been applied also to parallel algorithms in this way:

$$Stot(N) <= (S+P)/ (S+P/N)$$

where N is the number of processors. S and P are, respectively, the program fraction that has

to be run sequentially and that can be run in parallel; thus (S+P)= 1.
In this way, if there is even a very small serial fraction, the performance/cost ratio decreases quickly when the number of processors increases. If we hypothesize that the problem is infinitely parallel (i.e. S=0) then we obtain an upper limit to the achievable speedup in the best case: S(N)<=N

In 1988 Gustafson proposed a new speedup definition, called "scaled speedup".
Amdahl's law hypothesizes a fixed-size problem and a variable execution time. Instead Gustafson argued that a more realistic scenario would be getting the problem into proportion on the basis of the number of available processors. In Gustafson's model the execution time is fixed and the problem size is scaled when we use more powerful computers. It is possible to modify the problem size in many ways, among which, increasing spatial resolution, using a smaller time interval, adding more variables, etc. Gustafson revisits Amdahl's law starting from different premises. He uses S' and P' to represent the serial and parallel execution time on a parallel system. Because (S'+P')=1, and Tser= S'+P'xN, his speedup formula is:

$$Sscaled(N)= N +(1-N)xS'$$

According to Gustafson, if a massively parallel computation isn't efficient for a code-problem pair, it could be efficient for the same code on a larger problem size. The aim of parallelism is to maximize the throughput keeping a constant execution time.
For many applications having a less than linear speedup, i.e. S=N using 2N CPUs, is satisfactory given the lower and lower CPU cost. Although a theoretical boundary for the greatest obtainable speedup (linear speedup) exists, superlinear speedup effects are possible. The superlinear speedup term refers to a speedup greater than N on N CPUs. According to Amdahl's law this is impossible.

A superlinear speedup is possible only if the total work done by N CPUs is strictly less than the single CPU work. In the past, the main causes of superlinear speedup were the inefficient serial algorithm implementations [9][13]. In [8] some causes of superlinear speedup are explained, among which, the most interesting one is the growth of distributed cache. In [12] Gustafson asserts that the performance can improve if the problem size per processor is shrunk. He probes the case of fixed size problem and increasing CPU number[9]. On a distributed memory architecture the different access speed to memory levels can bring to superlinear speedup effects.
When the problem size goes beyond the RAM size, that performance plan part is labelled as "insufficient memory" (Figure 2). However there is always a mass storage that contains the problem even if it will be slower. In the same way when a problem reduces to few variables per processor, it could use the caches or registries rather than RAM. A superlinear speedup can show in this case when the model crosses the boundary region shifting to an higher speed plan sector. The speed increase has to be enough to cover the common causes of parallel inefficiency. Considering the memory hierarchy we find out that the efficiency doesn't always drop when processors increase.

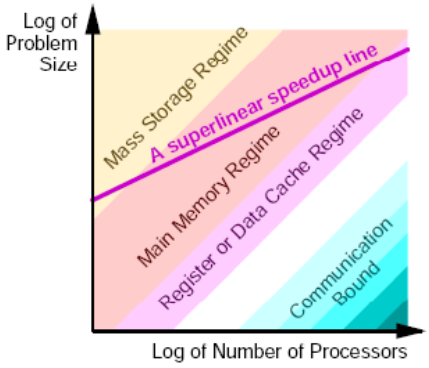


Figure 2 - Performance Plan

This speedup can be explained when we envisage a realistic computational model

considering full memory hierarchy and access time.

IV. ALGORITHMS USED DURING TEST

The performance evaluation has been carried out in terms of speedup, efficiency and scalability analysis of parallel algorithms using the MPI paradigm. Well known benchmarks have been tested: Matrix Multiplication [3], Bitonic Sort [4] and Pascal's Triangle algorithms [5]. These algorithms have been chosen because they belong to the three most common computational complexity classes: O(n), O(n*log(n)), $O(n^2)$. They have been coded in C and compiled using MPI libraries: MPICH, MVAPICH, MVAPICH2. The tests have been executed on the COMETA Grid infrastructure.

A. Matrix Multiplication Algorithm

Matrix multiplication is one among the linear algebra basic operations and consists of multiplying a matrix by either a scalar or another matrix. It is used in many benchmarks to evaluate the performances of different architectures and it has a predominant role as prestational index in high performance computing. The algorithm allows to modify the dimensions of the input matrixes. In particular, only the first matrix row number is modified. In this way, the resulting matrix computation complexity is linear (O(n)). This algorithm is implemented using two different communication strategies: point-to-point and collective. The serial algorithm does the simple row-column multiplication. Instead the parallel algorithm has a master-slave implementation.

B. Bitonic sort Algorithm

Bitonic sort is a sorting algorithm designed specifically for parallel machines. A sorted sequence is a monotonically non-decreasing (or non-increasing) sequence. A bitonic sequence is made up of two subsequences, the first monotonically non-decreasing and the other monotonically non-increasing (or viceversa).

The algorithm applies a "compare and exchange" mechanism that compares the elements in the same position inside each subsequence and exchanges the element if they aren't in the desired order. This action is applied recursively dividing each subsequence into two halves.

The algorithm performs a bitonic merge on the sequence, which gives us a bitonic sequence in each half and all the larger elements in the upper half. As explained in [6], the computational complexity is O(n*log(n)).

C. Pascal's triangle Algorithm

Pascal's triangle algorithm determines the coefficients which arise in binomial expansions. The single-processor algorithm performs a direct computation of Pascal's triangle. The algorithm is as follows: begin with a row of integers. Compute the next row of integers by adding pairs, in order, of immediately neighboring integers of the previous row. Continue until the desired height of the triangle, or length of propagation, is reached. The parallel algorithm instead evenly divides the row of initial integers into one block per processor. An important detail is that the trailing edge of one block will influence the leading edge of the next, a result of the nature of the interdependence in this system: interactions are localized. Nonetheless, work can be partitioned between the processors with their dependencies handled efficiently. This algorithm performs a frequent data exchange because every processor needs the boundary elements that are in neighboring domain. The computational complexity is $O(n^2)$, where n is the triangle's height, because the operations depend on the height of the triangle and on the last row elements number; the number of triangle elements are n*n/2.

V. RESULTS AND FINAL REMARKS

The tests have been executed on the COMETA Grid infrastructure. A different processors number, from 0 to 128, has been used. The

execution time has been measured using MPI_Wtime() function that returns a double-precision floating point number. The measure has been taken at the begin and at the end of the program. The master process collects the execution time values and is liable of merging data coming from the other processes. This overhead is accounted for in the measure.

For the matrix multiplication algorithm, we show in Figure 3 the results of two different communication strategies: point-to-point (caseA) and collective (caseB) implementation considering the smallest and largest size of the test data. The results have an overall "bell-shaped" trend, where the highest point corresponds to the optimal number of processors, beyond which further input data distribution is not convenient. The use of collective implementation guarantees better performance than point-to-point one. In both cases MVAPICH's performance resulted the best compared to the other implementations available. In fact, it offers minimal communication overhead allowing superior scalability on the infrastructure. When the input data size is small using a larger number of processor causes a speedup reduction due to the increased overhead that covers the parallelization advantages.

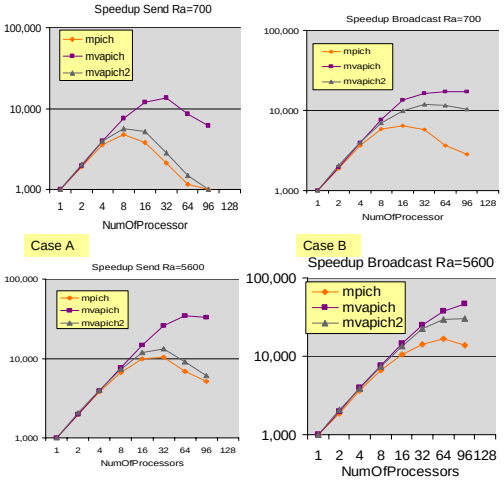


Figure 3 – Speedup of the Matrix Multiplication alg.

Instead, increasing the matrix size, the speedup increases and the optimal number of processors is higher. In this case, according to Gustafson, an algorithm is scalable when, as the problem size and the number of CPUs grow, the execution time is fixed. If a massively parallel computation isn't efficient for a code-problem pair, it could be efficient for the same code on a larger problem size. We can see in Figure 4 that the algorithm is scalable up to 16 CPUs, after that there is a remarkable communication overhead that covers the benefits of parallelism.

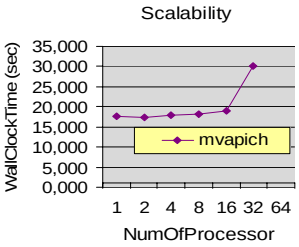


Figure 4 – Scalability of Matrix Multiplication Alg.

In the case of bitonic sort, we show in Figure 5 the speedup and scalability of the algorithm. To obtain a measurable execution time we used an input vector of 2^{27}~135MB. So, because we do not have 135 available CPUs, there is no superlinear speedup effect. This algorithm often performs exchanges between the vector data and this is the reason that MPICH has the worst performance. In this case the InfiniBand communication network is needed to avoid the communication channels saturation.

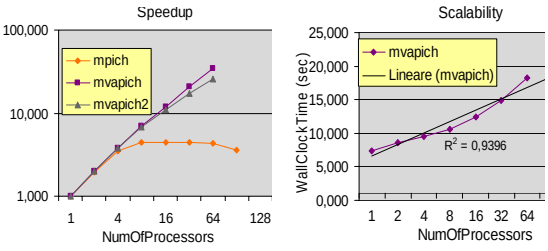


Figure 5 – Speedup & Scalability of the Bitonic Sort alg.

The scalability curve has a realistic trend based on the computational complexity of the algorithm, O(n*log(n)). In this graph with lin-log axis, the curve is close to a straight line. R-squared index is 0.94. It is high enough, but if we consider the only data values up to 16 CPUs, it raises to 0.97. This value changes because the communication overhead is not taken into account in the model and starts being relevant from 16 CPUs onwards.

Considering the Pascal Triangle's algorithm, we show the effect due to the Grid distributed architecture. Figure 6 shows superlinear speedup effect and efficiency value higher than one. Employing a higher number of processing units the available main memory grows, but also the cache memory does.

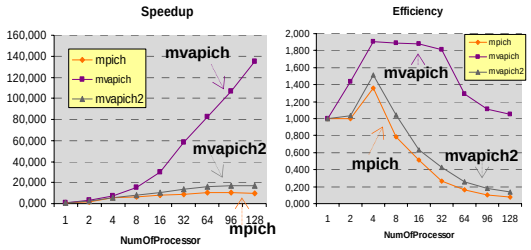


Figure 6 – Speedup & Efficiency of the Pascal Triangle alg.

If the problem size fits into the aggregated cache the resulting speedup can be superlinear. In particular, using 4 CPUs there is an efficiency peak in all the algorithm implementations because the data are completely stored in the local cache.
Let's focus on the MVAPICH implementation because is the best one, and let's test different problem sizes (i.e. 600KB, 1.2MB, 2.4MB, and 4.8MB).

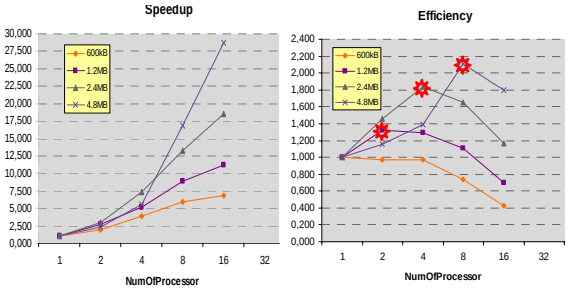


Figure 7 – Superlinear Speedup & Efficiency

The bigger the problem the higher superlinear speedup achievable. Figure 7 shows the speedup effect due to different data distribution on a growing number of resources. Given a problem size we can find the number of resources that will yield superlinear speedup.
L2 cache size per CPU in the COMETA Consortium Grid infrastructure is 1MB. If the problem size fits in a single processor L2 cache (i.e. 600KB) there is no superlinear effect; as the problem size grows the peak point will move to the right, in correspondence to the minimum number of CPUs that can hold the whole problem in their aggregated cache.

The performance evaluation analysis allowed to highlight the advantages in the use of low latency and wide band communication network InfiniBand compared to Gigabit Ethernet; higher parallel speedup using InfiniBand over Gigabit Ethernet has been achieved. In addition MVAPICH has been proved to be the most efficient MPI implementation compared to the others available as it offers minimal communication overhead allowing superior scalability on the infrastructure. Finally, some algorithms have shown superlinear speedup effect due to a different distribution of data on a growing number of resources. On a distributed memory architecture the different access speed to memory levels can bring to superlinear speedup effects. This speedup can be explained when we envisage a realistic computational model considering full memory hierarchy and access time.

ACKNOWLEDGMENTS

This work makes use of results produced by the PI2S2 Project managed by the Consortium COMETA, a project co-funded by the Italian Ministry of University and Research(MUR) within the Programma Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta

Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] SM InfiniBand. Trade Association. InfiniBand Architecture Specification Volume 1, Release 1.0, 2000.
[2] http://www.mpi-forum.org/
[3] https://computing.llnl.gov/tutorials/mpi/samples/C/mpi_mm.c
[4] http://www.tools-of-computing.com/tc/CS/Sorts/bitonic_sort.htm
[5] http://daugerresearch.com/vault/parallelpascalstriangle.shtml
[6] http://www.iti.fh-flensburg.de/lang/algorithmen/sortieren/bitonic/bitonicen.htm
[7] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A High-Performance, Portable Implementation of the MPI, Message Passing Interface Standard. Technical report, Argonne National Laboratory and Mississippi State University.
[8] Helmbold, D. P. and McDowell, C. E., "*Modeling Speedup(n) greater than n*" 1989 International Conference on Parallel Processing Proc., (1989), Volume III, pp. 219-225.
[9] J. Gustafson, "*Fixed Time, Tiered Memory, and Superlinear Speedup*," in Proc. of the Fifrh Conference on Distributed Memory Computing (DMCCS), 1990.
[10] Jiuxing Liu, Jiesheng Wu, Sushmitha P. Kini, Darius Buntinas, Weikuan Yu, Balasubraman Chandrasekaran, Ranjit M. Noronha, Peter Wyckoff and Dhabaleswar K. Panda "*MPI over InfiniBand: Early Experiences*".
[11] MUNIRA HUSSAIN, RINKU GUPTA, AND TONG LIU. "*Using OpenFabrics InfiniBand for HPC Clusters*"
[12] Gustafson, J. L., "*Reevaluating Amdahl's Law,*" Communications of the ACM, Volume 31 (1987), pp. 532–533.
[13] V. Faber, O. M. Lubeck, and A. B. White, Jr., "*Superlinear speedup of an efficient sequential algorithm is not possible*," Parallel Computing, vol. 3, pp. 259–260, July 1986.

Dott. Simona Blandino – She was born in Ragusa (Italy) in 1982. She received her "laurea specialistica" Master degree in Informatic Engineering from the University of Catania in July 2008. Actually, she works as System Administrator for Consortium COMETA.

Prof. Salvatore Cavalieri - He was born in Catania (Italy) in 1965. He received his "laurea" degree in Electronic Engineering from the University of Catania in 1989. In 1993 and 1995, he received a PhD in Electronic and Computer Science Engineering, and a post-PhD in Electric Engineering from the same University. In 1995 and 1998 he became Researcher and Assistant Professor, respectively, of Computer Science at the University of Catania. Actually he is Full Professor of Computer Science at the same University, Faculty of Engineering, Department of Computer Science and Telecommunications Engineering. He has been involved in many national and European research projects, and for some of them he held the coordination.

VisualGrid: On-Line Video Streaming for Application Control and Demonstration

Giuseppe Andronico[1], Roberto Barbera[1,2],
Marcello Iacono Manno[3], Giuseppe La Rocca[1]

[1]INFN Catania – Via S. Sofia 64, I-95123 Catania, Italy
{giuseppe.andronico, giuseppe.larocca}@ct.infn.it
[2]University of Catania – Via A. Doria, 6, I-95125 Catania, Italy,
roberto.barbera@ct.infn.it
[3]Consorzio COMETA, Via Santa Sofia 64, I-95123 Catania, Italy,
marcello.iacono@ct.infn.it

***Abstract*— Several Grid applications produce images that can be collected and encoded as a unique video to be visualized also when the job that produces the images is still running. This feature maybe helpful for both demonstration purposes or to control job evolution. Here the VisualGrid on-line streaming tool is presented in the two available forms, either as a single script to be integrated into the producer application, or as a dedicated service of the Genius web portal.**

***Index Te1ms*— Grid Computing, on-line streaming tool, job control.**

I. INTRODUCTION

Grid technology, based on opens standards and protocols, is the technology which allows user to share a wide heterogeneous of distributed computational resources such us: supercomputers, storage systems, data resources and instruments, regardless of their geographical location. The new paradigm introduced by this new technology has become extremely important for many scientific and industrial applications belonging to different domains such as Astronomy, Computational Chemistry, Earth Science, Financial Simulation, High Energy Physics and Biomedicine as well. In this scenario, today we have an increasing set of grid applications, especially those producing complex output patterns after a long run (e.g. Fluent[1]), which can be easily shown to a wide audience and monitored with the strong impact of multimedia communication. In this paper the technical details of the VisualGrid on-line streaming tool to address this goal is shown.

A. The VisualGrid service

In this section we describe the design and the prototype implementation of this new service (VisualGrid) implemented on board of the GENIUS Grid Portal in order to permit non expert users to control jobs evolution using a web based interface. With this new grid service and the strong impact of multimedia communication the user's jobs can be better monitored and controlled. The design and implementation of VisualGrid has been achieved working in two different steps. First and foremost the distributed Grid infrastructure of the PI2S2 project[2] has been configured with a list of additional software packages necessary during the computation to generate and send the video in streaming to the user. The complete list of additional packages deployed in the infrastructure includes the following software: *VideoLAN[3]*, *FFMPEG[4]*, *Mplayer MENCODER[5]*, *GFALl C API[6]* and *Xvfb[7]*. Beside, a new tag, with the VLC-0.8 label, has been

published by the top BDII for the given VO in order to allow the WMS to identify only grid resources properly configured to provide this additional feature. Once the grid infrastructure was set up, some web services have been designed in the GENIUS Grid Portal[8][9] to make the service accessible to all the VO members. This computing web portal, powered by EnginFrame[10] provides to the end-user a transparent and abstraction layer to access the distributed services and resources of a typical grid infrastructure.

B. The GENIUS Architecture and EnginFrame

EnginFrame is a Web-based technology, developed by the Italian company Nice[11] S.r.l., that enables the access and the exploitation of Grid-enabled applications and infrastructure. It allows users, grouped within a VO, to exploit the whole power of a grid infrastructure hiding all its complexity to the end-users and providing an abstraction layer for the grid technologies. *EnginFrame* is also well known in the Grid research world for being the technology which *GENIUS (Grid Enable web eNvironment for site Independent User job Submission*)[12], the official Grid portal of the European project EGEE is based on. Due to the modularity and flexibility of EnginFrame, the Genius Portal can be easily customized to interact with several Grid component and/or VO allowing scientists to access, execute and monitor their own applications only by using a simple web browser. In this way the complexity of the underlying Grid infrastructure is hidden to the end users.

C. Design and implementation of VisualGrid.

The Service Definition Files (SDF™) are the core of the EnginFrame framework. Basically they are simple high-level XML files which describe how to link the existing command-line world to users' Web interface. Each SDF must have an *.xml* extension and, in order to be processed, must be included in the DOC_ROOT of the Web Server. In this work one of these SDF, based on EnginFrame XML language, has been created to enhance the EnginFrame framework with this new service. Two different implementation of this service have been considered as shown in figures 1 and 2

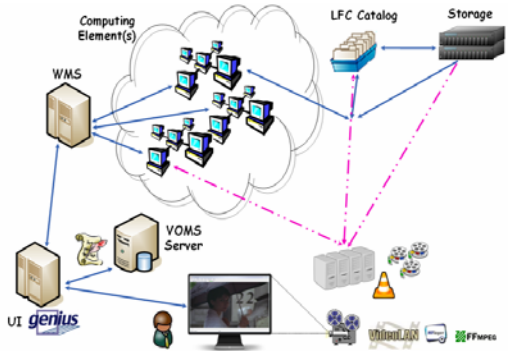


Fig 1 – Generation of the video output submitting a simple job.

Figure 1 shows the application workflow designed to generate, submitting a single job, the video output starting from the images generated by a produce application. The client side is represented by a user's workstation running a web browser. The server side is represented by a gLite User Interface (UI), equipped with the latest gLite middleware services to submit jobs and manage data in Grid, the Apache Web Server, the Java/XML portal framework EnginFrame developed by NICE Srl and the GENIUS Grid Portal itself. After the user has logged in, using the portal the user can provide the input settings before running the service (e.g.: frame rate, video output format, grid location of the image tar produced by the producer application on the LFC File Catalog).

Fig 2 – VisualGrid input settings.

When the service is started, the job running in the WN downloads from the SE the image tar produced by the producer application during its run and the MPlayer MENCODER. The C gFAL API have been adopted in order to speed up the downloading process of files from the SE. The image tar downloaded from the SE is processed by MENCODER and an output video will be created. Before the end of computation, an UDP network connection on port 1234 is initialized between the WN where the service is running and the public IP of the user desktop. This UDP connection will be used by VideoLAN software running in the WN to stream the video over the network to the user desktop.

Figure 3 depicts the application workflow designed to generate the video output with a Direct Acyclic Graph (DAG). In this case, depending by the number of images processed by each job, a DAG (Nx1) will be created on-the-fly and submitted by the portal.

All the video chunks reconstructed by each node of the DAG will be finally collected by a *Collector job*. This job is in charge to start the video streaming towards the user. In order to improve the performance of the service in this case, the following additional attributes have been considered: *Max_Running_Nodes* and *NodesCollocation*.

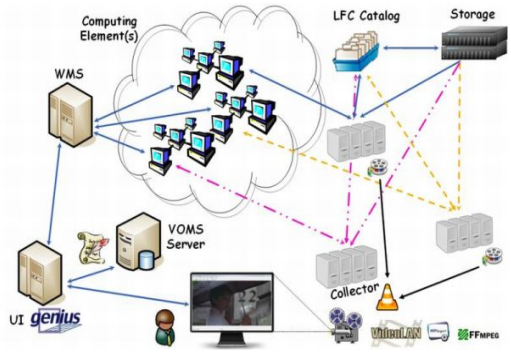


Fig. 3 - Generation of the video output submitting a DAG.

The first one indicates the maximum number of nodes of DAG that can be simultaneously submitted by DAGMan This attribute has been set to N, where N is the number of nodes of the DAG. The second one has been set to *true* to avoid the WMS submit the nodes of the DAG on different CE. We need this attribute because the chunks produced by each node of the DAG will be transferred as inputs by the Collector job.

VisualGRID service has been built on top the new EGEE gLite middleware and make use of many gLite services: the LFC File Catalog and Data Management System, the VOMS server, Storage Elements, Computing Element, User Interface and the Workload Management System as shown in the figures 1 and 2.

II. Conclusion

The on-line streaming tool introduced in this paper embedded in the Genius Grid Portal allow users to better monitor applications which can create, after a long run, complex output patterns. In this case the output of the Grid calculation can be easily shown to a wide audience of users thanks to the strong impact of multimedia communication.

Acknowledgments

We gratefully acknowledge all the people who supported this work contributing with ideas,

requirements and feedback.

REFERENCES

[1] Fluent – www.fluent.com
[2] The PI2S2 Project – www.pi2s2.it/
[3] VideoLAN – Free and Open Source software and video streaming, www.videolan.org/
[4] FFMPEG – ffmpeg.org/
[5] Mplayer – www.mplayerhq.hu/
[6] Data Access with GFAL – wiki.egee-see.org/index.php/Data_Access_with_GFAL
[7] Xvfb – www.xfree86.org/4.0.1/Xvfb.1.html
[8] EnginFrame – www.enfingrame.com/
[9] NICE Srl – www.nice-italy.com/
[10] Andronico G, Barbera R, Falzone A, Lo Re G, Pulvirenti A, Rodolico A – "GENIUS: a web portal for the grid" – Nucl. Instrument and Methods in Phy. Res. A 2003. Visit also the official GENIUS web site: https://genius.ct.infn.it/
[11] Barbera R, Falzone A, Ardizzone V, Scardaci D – "The GENIUS Grid Portal : Its Architecture, Improvements of Features ,and New Implementations about Authentication and Authorization" – Enabling Technologies: Infrastructure for Collaborative Enterprises, 2007. WETICE 2007. 16th IEEE International Workshops on.
[12] The GENIUS Grid Portal – https://genius.ct.infn.it

Grid Computing for Internet Traffic Engineering based on OSPF/IS-IS Protocols

[1]S. Carcangiu, [1]A. Fanni and [1]A. Mereu
[1]*Department of Electrical and Electronic Engineering,*
University of Cagliari, Cagliari, Italy,
{s.carcangiu, fanni, anna.mereu}@diee.unica.it

***Abstract* — An original Grid Computing enabled implementation of an optimization tool for optimal traffic routing in telecommunication networks is proposed in this paper. The main objective of the tool is to find the optimal link weights in order to avoid congestion and to guarantee full survivability of the network. A meta-heuristic technique, based on Tabu Search, is proposed here to solve it. The optimal design of a real backbone network of an Italian ISP is demonstrated in this paper.**

***Index Terms*— Traffic Engineering, IS-IS Protocol, Optimization methods.**

I. INTRODUCTION

IN a telecommunication network, the IP flows are forwarded by routing protocols that are able to find the shortest paths for each flow by associating a metric (weight) at each link, and by computing the paths by applying the Dijkstra algorithm. In case of multiple equivalent shortest paths, the traffic will be evenly split. In backbone networks, the weight associated to each link can be changed by the network operator by means of his experience and, in most of cases, without taking into account the Quality of Service (QoS). The weights could be set proportional to the physical lengths of the links, but often the recommendations of major router vendors is to set the weight of a link inversely proportional to its capacity.

In the last few years, several works have been presented which aim at optimizing the traffic flows within a network.

In [1] and [2] a meta heuristic approach is presented whose objective is to find the optimal OSPF weights set that maximizes the routing performances under normal working condition. In [3], the Simulated Annealing algorithm is used to optimize the OSPF link weights and, as further improvement of the QoS, a small number of MPLS paths is found using two mixed-integer programming models. In [4], an adaptive load balancing methodology in OSPF-based networks is presented. In the proposed approach the routers, knowing the measured link loads, hold the capability to make independently small changes in the load distribution by adjusting their own traffic splitting ratios. All the mentioned papers do not take into account the problem of the survivability of the network in case of failure. In [5], a Tabu Search algorithm is proposed for assigning link weights in IS-IS/OSPF networks taking into account the single failure of any possible link in the backbone. The authors claimed that it is not possible to find a single weight set that performs equally well both for transient failures and in the absence of failures. Nevertheless, the algorithm finds a weights set that restricts link loads to less than 90% during any link failure.

In this paper, we extend the work done by [6] by presenting the parallel implementation of the algorithm in a computational Grid, in order to limit the computational time required by the algorithm to find the optimal solution.

In the algorithm, the main objective is to find the optimal IS-IS/OSPF link weights set in order to achieve robust and economically efficient solutions for the optimal traffic routing in the network, minimizing the risk of

wrong decisions in presence of congestions deriving from critical situations such as uncertainty of demands and transitory or enduring failure conditions. This problem is NP-hard [7], even for rough approximation, and we have developed a meta heuristic technique, based on Tabu Search, to solve it.

The paper is organized as follows. In Section II an overview of the optimization algorithm and the Tabu Search meta-heuristic are presented; in Section III its implementation in the Grid infrastructure is described, while in Section IV the results are reported. Some conclusions will end the paper.

II. SEQUENTIAL OPTIMIZATION ALGORITHM

It is often convenient to introduce graph models for representing the topology of a back bone telecommunication network, because a number of efficient techniques is available for handling them.

Let $G=(N,A)$ be a graph representing the network, where N is the node set and A is the arc set. Let $\boldsymbol{M}$ ($\mu = |\boldsymbol{M}|$) denote a set of origin-destination pairs, or *commodities*; for each $m \in M$, we will denote by $s(m)$, $t(m)$ and d^m respectively the origin node, the destination node and the traffic demand of the commodity m. For each non-zero demand d^m the routing problem consists on distributing the requested flow over paths from s to t.

Since real information is extremely difficult to obtain, as the traffic in the Internet varies with continuity in the short term, it is necessary to consider a point of operation on a long time basis. This point is represented by the traffic demands matrix and its definition is very important because it could alter the optimization process. The demand matrix could be based on concrete measures of the flow between source-destination pairs or it could also be based on a concrete set of customer subscriptions to virtual leased line services [1].

The optimization problem under investigation consists of finding a set of IS-IS/OSPF weights that allows the network to satisfy the demands of the commodities in $\boldsymbol{M}$ in such a way to minimize the maximum *link congestion* of the network, under survivability constraints. Here the congestion of an arc (i,j) is defined as the total flow sent along (i,j) over the associated capacity c_{ij}.

Note that, defining *link congestion* as the maximum of the congestion of the two directed arcs associated with that link, link congestion and arc congestion coincide.

The target is to guarantee a 100% level of survivability in case of single link failure on logical topology, which is the most probable fault event. In particular, as reported in [8], for the unplanned failures, 70% of them affect a single link at a time, while 30% are shared by multiple links and can be attributed to router-related and optical equipment-related problems.

A network is said to be survivable if it is capable to reroute all its flow under the failure of any of its edges.

The proposed method performs an off-line optimization and makes use of the information deriving from the observation of the traffic flowing in the network.

Different scenarios have to be considered that are the normal working condition scenario plus the failure condition scenarios. Starting from an initial weights configuration, (e.g., with unitary weights), the algorithm, for each scenario, finds the shortest path(s) for each nodes pair and compute all the links load. The maximum is selected as value of the objective function. The meta-heuristic Tabu Search (TS) strategy is used to change the link weights in order to find the minimum of the objective function. TS performs the search for the optimal solution by exploring the variable space performing suitable *moves* and storing the features that correspond to bad previous moves. Such features are labeled as tabu and

stored in the so called Tabu List: in this way they are avoided during the search for the optimum. This method does not rely on analogy with physical systems, like for example Genetic Algorithms [9] or Simulated Annealing [10], and many implementations are mostly or totally deterministic. This is because TS is based on the incorporation of adaptive memory and responsive exploration in order to qualify problem solving as intelligent. The use of adaptive memory contrasts with "memory-less" designs such as those previously mentioned, and with "rigid memory" designs, such as those exemplified by branch and bound. The emphasis on responsive exploration derives from the supposition that a bad strategic choice can yield more information than a good random choice.

In TS methods the choice of the move is of crucial importance, i.e. the modification rule that allows passage from one configuration to another. At each iteration, the neighborhood associated to the current configuration is built performing all the possible moves that can be played starting from that configuration.

The simple iterative scheme of the TS is enhanced introducing several rule-of-thumb criteria such as aspiration (an element may be removed from the Tabu List under certain conditions), intensification (deeply exploration of a region looking promising), and diversification (leave a region that does not look promising).

In our problem, each configuration of the back bone network is completely defined when link weights have been chosen. In fact, the IP demands are forwarded by the IS-IS/OSPF link state routing protocol, which computes the paths of any commodity by applying Dijkstra algorithm [11].

At each iteration of the TS, the neighborhood associated to the current configuration is built performing all the possible moves that can be played starting from that configuration by changing one values of the string at a time in the current configuration. We will call these moves Cartesian moves.

In order to introduce the survivability constraints, different scenarios have to be considered that are the normal working condition scenario and the failure condition scenarios. Starting from an initial weights configuration, (e.g., with unitary weights), the proposed TS algorithm, for each scenario, finds the shortest path(s) for each nodes pair and compute all the links load. The maximum is selected as value of the objective function. The algorithm is composed of the following steps:

1. *Set an initial weights configuration*
2. *Store the current solution as best solution*
3. ***for** each scenario **do***
   a. *compute shortest paths*
   b. *compute link utilization*
   c. *select the maximum link utilization*
4. *Find the maximum utilization among all scenarios*
5. *Change the weights set (Tabu Search)*
6. ***if** the current solution is better than the best solution*
   a. *Upgrade the best solution*
7. ***if** stop criteria is not reached*
8. ***return** to step **3***
9. ***else** return the best visited solution and the corresponding optimized weights set*

As previously cited, a move consists in changing the value of one weight at a time. The new value will range in the IS-IS/OSPF weights range or in a given sub range, whose size is a design parameter. In order to limit the computation time, the size of the neighborhood can be changed according to the network size. To avoid any loss of generality, we adopted a strategy in which the sub range dynamically varies in a given interval [min, max] following a Reactive TS scheme [12].

III. Implementation of Tabu Search in the Grid Computing Infrastructure

The schema of the TS algorithm is inherently sequential, but it can be easily parallelized. In fact, the most important requirement in using a TS algorithm consists of defining the set of admissible moves, i.e., in defining the neighborhood set of a given solution. As previously mentioned, our TS implements Cartesian moves, which consists of changing the value of one design parameter at a time. The exploration of a single variable can be carried out by independent jobs, which means that for each configuration in the neighborhood, the evaluation of the objective function has to be performed. It is important to notice that in the Grid the execution of cyclic jobs is not allowed. This leads us to design the core of our algorithm to run locally in the user interface and to run the parallel exploration of the neighborhood in the Grid. This has been possible by splitting the program in a master-slave architecture. Throughout Application Programming Interfaces (APIs), the master program launches the execution of the slaves. To each slave a different variable to explore is assigned. When the jobs are done, the master program analyzes the entries written on a database of metadata by the slaves and performs the optimal current move. The interactions between the master program, the slaves and the database are summarized in Figure 1.

A. Master Program

At the beginning, the master program sets up a database that is needed to obtain the Tabu List that keeps trace of the previously explored configurations. As we previously explained, this is due to the idea of avoiding a cycling around an already explored configuration. To do this, the master program initializes a database to which also the slaves can have access. Each slave writes its own best solution into the database: when all of them finish writing, the master program reads and compares the data looking for the optimal move. After the creation of the tabu list database, a while cycle starts. It is governed by two parameters: the total number of calls to the objective function and the number of iterations of the TS that we want to perform. It is also possible to stop the algorithm when the objective function reaches a certain threshold. During the first iteration the master sends to the slaves the jobs by starting from an initial configuration. The job is a parametric job where all the slaves have to do the same things, that is, given the current configuration, they explore the neighborhood of one variable. The parameter that changes among the slaves is the variable to consider. After submitting the parametric job, the master asks the Logging and Bookkeeping component about the status of the job. When the job is done, the master program analyzes the entries written by the slaves and finds the optimal configuration. The optimal configuration has to be sent to the slaves during the following iteration. This is accomplished by writing the optimal configuration in a file and passing it to the slaves.

B. Slave Jobs

The slaves are stand alone executables that perform the TS algorithm exploring the neighborhood of a single variable. When the slave finds a configuration that is a good candidate to be stored in the database, it checks

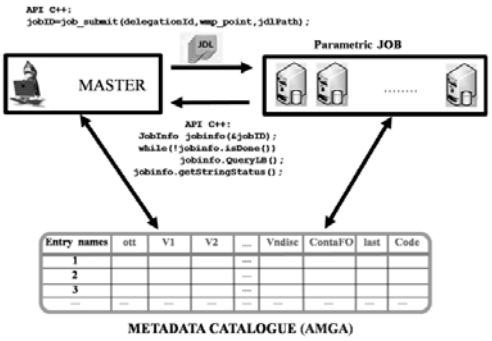


Fig. 1. Implementation of the algorithm in the Grid computing Infrastructure.

the tabu list: if that configuration is not already stored in the list, the slave stores the new configuration in the database. As we explained in the Section II, in our application, the variables correspond to the link weights of the network. For this reason the computation of the objective function consist in computing, for each failure scenario, the shortest paths, the consequent links utilization and in selecting the maximum link utilization.

IV. RESULTS

In this section we will show the results obtained by applying the TS optimization algorithm to a real network. The network under study is a real backbone network of an important Italian ISP (Fig. 2).

In this case we have computed the maximum link utilization among all the single link failure scenarios.

Table I
Real Backbone Results, Survivability

Network Configuration	Maximum Utilization
Unitary Weights	128% of link capacity
ISP Weights	117% of link capacity
Optimized Weights	85% of link capacity

As shown in Table I, the results obtained from the optimization process have been compared with those of the unitary weights configuration and with those deriving from the current ISP configuration (Fig. 3). With the optimization of the link weights, the maximum utilization decreases of more than the 27% with respect to the initial ISP configuration of the weights (Fig. 4).

The performances of the Grid-enabled TS are reported in Table II in terms of the speedup and efficiency parameters usually used to evaluate the performances of parallel algorithms [13].

As the Grid used is composed by a number of identical CPUs, we can extend the concept of speed up and efficiency, used in parallel systems, in the grid environment. The speedup is defined as the ratio of the execution time of the algorithm when executed on one processor

Table II
Grid-Enabled TS versus Sequential TS

p	T_p[s]	S_p	E_p
1	80,33	1	-
4	33.3	2.41	60%
8	24.34	3.3	41%
16	19.1	4.2	26.2%
20	16.4	4.9	24.5%
32	15.1	5.3	16.5%
40	14.6	5.5	13,7%
80	13.8	5.8	7%

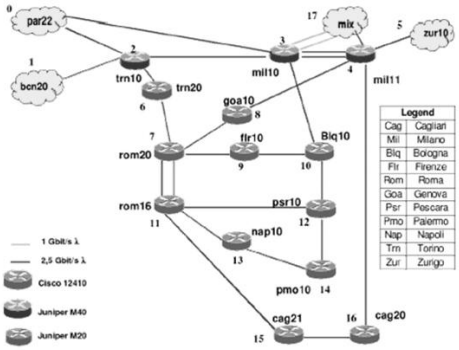


Fig. 2. Italian portion of the Tiscali International network.

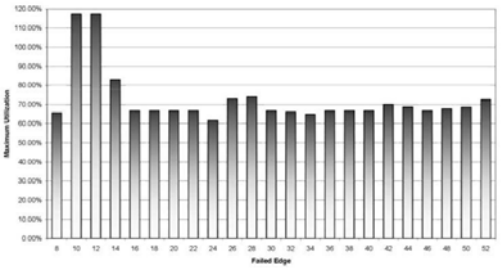


Fig. 3. Maximum link utilization with the actual weights.

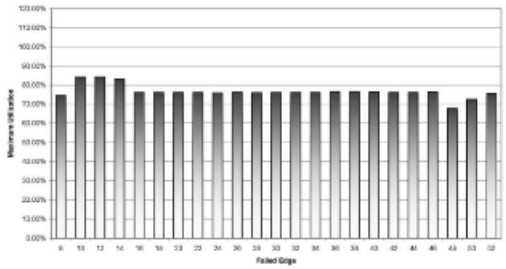


Fig. 4. Maximum link utilization with the optimized weights.

to that when executed on **p** processors: $\mathbf{S_p = T_1/T_p}$, whereas the efficiency is defined as the ratio of the actual speedup versus the theoretical maximum speedup equal to **p**. Here $\mathbf{T_1}$ and $\mathbf{T_p}$ are the total times required to run one iteration of TS on a single CPU, and on p CPUs respectively. The maximum efficiency is obtained when the speedup is the closest to the theoretical speedup.

Note that the algorithm in Grid environment has speedup limits and appears to saturate due mainly to communication overhead, and to some uncertain factors as instability of computing nodes, dynamic change of Grid environment and total load on the Grid.

V. Conclusion

In this work the implementation on a Computational Grid of an innovative algorithm to guarantee survivability of networks using IS-IS/OSPF routing protocol is presented. Results show that an optimal link weights set avoids the risk of congestion or deny of service in case of single link failure. The performed experiment showed that a degree of overhead is introduced by the porting of the application over the Grid. This is mostly due to the communication delay between the computing elements. This has to be taken into account when designing the porting of an application in computational grids. The overhead can be neglected if the parallelized jobs have a high computational load.

Acknowledgments

This work makes use of results produced by the Cybersar Project, a project co-funded by the Italian Ministry of University and Research (MUR) within the "Programma Operativo Nazionale 2000-2006". More information is available at http://www.cybersar.it.

References

[1] B. Forzt, M. Thorup, "Internet Traffic Engineering by Optimization OSPF Weights," in *Proc. 19th IEEE Conf. on Computer Communications (INFOCOM)*, 2000, pp. 519-528.

[2] B. Forzt, M. Thorup, "Increasing Internet Capacity Using Local Search," *Computational Optimization and Applications.* Belmont, 29(1), 2004, pp. 13-48.

[3] A. Riedl, "Optimized Routing Adaptation in IP Networks Utilizing OSPF and MPLS," in *Proc IEEE ICC.*, 2003.

[4] R. Susitaival, S. Aalto, "Adaptive load balancing with OSPF," In *Proc. 2nd Int. Conf. On the Performance Modeling and Evaluation of Heterogeneous Networks (HET-NETs 04)*, 2004.

[5] A. Nucci, B. Schroeder, S. Bhattacharya, C. Diot, and N. Taft, "IGP link weight assignment for transient link failures," In *Proc. 18th ITC*, 2003.

[6] D. Cherubini, A. Fanni, and C. Murgia, "IS-IS/OSPF weights optimization for survivable backbone networks: a Tabu Search meta heuristic approach," in *OPNETWORK 2006*, Washington DC , 2006.

[7] P. Fekete, W. Hochstattler, S. Kromberg and C. Moll "The complexity of an inverse shortest path problem,", in *Contemporary Trends in Discrete Mathematics: From DIMACS and DIMATIA to the Future* ,1999, pp.49-113.

[8] A. Markopolou, G. Iannaccone, S. Bhattacharyya, C. Chuah, C. Diot, "Characterization of Failures in an IP Backbone," In *Proc. of INFOCOM*, 2004.

[9] D.E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning," Kluwer Academic Publisher, 1989.

[10] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, New Series, Vol. 220, No. 4598., 1983, pp. 671-680.

[11] E. W. Dijkstra, "A note on two problems in connexion with graphs," In Numerische Mathematik, Vol. 1, No. 1., 1959, pp. 269-271.

[12] F. Glover, M. Laguna, "Tabu Search," Kluwer Academic Publisher, 1997.

[13] F. Luna, A. J. Nebro, E. Alba, "Observations in using Grid-enabled technologies for solving multi-objective optimization problems," *Parallel Computing*, vol. 32, pp. 377-393, 2006.

Sara Carcangiu received her Dr. Eng. Degree in Electronic Engineering on December 2003 from the University of Cagliari. On January 2006 she started her PhD at the University of Pisa.

Her primary research interests are related to multi-objective optimization.

Alessandra Fanni received the laurea degree in Electrical Engineering from University of Cagliari, Italy, in 1981. She is presently full professor of electric circuit theory in the Department of Electrical and Electronic Engineering of the University of Cagliari. She co-ordinates several research projects and she is the scientific responsible of several research contracts for young researchers and PHD students. Her research interests includes the following main areas: Circuit Theory, System and Signal Theory, Combinatorial Optimization Algorithms, Numerical methods for electromagnetic fields and electromagnetic compatibility, Neural networks.

Anna Mereu received the MSc degree, with honors, from the University of Cagliari, Italy, in October 2006.

In 2007 she joined the Department of Electrical and Electronic Engineering at the University of Cagliari, Italy, where she is now a PhD student of the research group of Circuits Theory. Her research interests includes the Linear Programming and Meta-heuristics techniques for the optimization of traffic flows in telecommunication networks.

Grid2Win: the Grid for Microsoft Windows

Elisa Ingrà[1], Roberto Barbera[1,2] , Dario Russo[2], Fabio Scibilia[2]

[1]*INFN, Catania, Italy, {elisa.ingra, roberto.barbera}@ct.infn.it*
[2]*Consorzio COMETA, Catania, Italy, {dario.russo, fabio.scibilia}@ct.infn.it*

***Abstract*— Grid Computing has several meaning in the distributed computing ("Multiple independent computing clusters which act like a Grid", "The creation of a supercomputer by using spare computing resources within an organization", etc…..), but as a matter of fact the Grid paradigm would be a natural step in the evolution of the High Performance Computing (HPC). Nowadays many research domains are capitalizing on Grid computing, however, to be able to properly use the current Grid infrastructures, usually it is required that user has advanced skills on Unix based systems. Even if the Linux distributions are in full growth, many users still prefer to use Microsoft Windows. The Grid2Win project aims at "opening" the Grid services to Microsoft applications and integrating MS Windows clusters into existing Grid e-Infrastructures. Grid2Win is a Cygwin (Microsoft Windows platforms) port of the entire gLite suite. It allows to access the Grid under Microsoft operating systems and enables a farm to run jobs with native Windows application using a ported TORQUE version, on a Windows farm transparently to the gLite multicluster interoperability. Moreover Grid2win offers a user-friendly Graphical User Interface (GUI) to access Grid common facilities. GUI is really easy to use as any Windows based applications. There are other cross platform solutions like GANGA, but they are only for User interface; Grid2Win is native gLite implementation rebuilt under Windows. There are no projects other then Grid2win supporting a gLite windows farm.**

***Index Terms*— Grid, Microsoft Windows, gLite, Graphical User Interface, porting, command line interface.**

I. INTRODUCTION

NOWADAYS many scientific communities make use of distributed computational power spread through network by the Grid [1][2][3] in order to elaborate their own data and make new contribute to research in general. We can mention the LHC Experiment [4], that is expected to produce roughly 15 PB of data annually – enough to fill more than 1.7 million dual-layer DVDs a year! – and thousands of scientists around the world want to access and analyse these data. Usually the interested researchers are physicists, biologists or archaeologists that are users without, or with a little computer knowledge. Even if we are in the Internet era, many people do not have much familiarity with computers. Above all, many Grid users do not know very well, or do not feel comfortable with the command line of Linux shells that is the "front door" for the Grid services. So, in order to make easier their approach to the Grid and its computational power, many projects are carrying out several enhancements on Web Portals (more user friendly than a Linux console) or generic User Interfaces. Moreover many Grid users prefer Microsoft Windows as their own operating system even if Linux distributions are in full growth.

In this article we present the Grid2Win project conducted at the INFN [5] (the Italian Institute for Nuclear Physics) of Catania. Aim of the project is to have gLite technology running over Windows platforms. The gLite [6] middleware is a set of software services to build up Grid infrastructures. However, the current gLite implementation has been developed to be deployed onto Scientific Linux (SL) machines as a part of the EGEE (Enabling

Grids for E-science in Europe) project [7]. In order to migrate its external behaviour from Scientific Linux to Windows platforms, we have rebuilt all the services from original sources and made little modifications. Moreover, with the introduction of the new Microsoft Compute Cluster Server (CCS) [8], we integrated it into gLite as a new type of Windows-native local scheduler. In chapter II we will give describe the gLite middleware. In chapter III we will introduce the Grid2Win project and discuss results obtained up to this writing, while in chapter IV we will discuss conclusions and future works.

II. GLITE MIDDLEWARE

The gLite middleware is a collection of software tools and services developed within the EGEE project to build Grids over the GEANT network, which interconnects many research institutes across Europe. gLite is built on top of Globus Toolkit (GT) [9] [10] to solve issues related to the security of Grid, data movement as well as management and many other aspects.

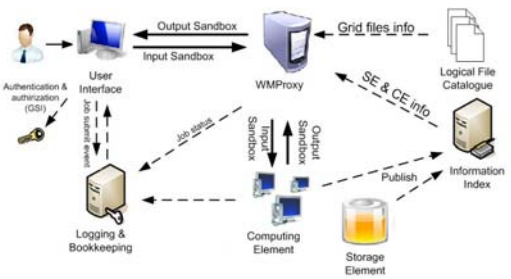


Figure 1. gLite Architecture

Figure 1 shows the architecture of gLite as collection of interacting services, essentially they are:

- The Security Services manage Grid credentials of the user and log in to the system following the Single Sign On (SSO) principle via the Globus Security Infrastructure (GSI);
- The Workload Management Services submit, monitor, cancel a job or retrieve its output;
- The Data Management service copy, remove, move and replicate Grid files from resource to resource;
- The Information Services browse the Information System and obtain data about deployed Grid resources and services.

Users access to the Grid via the User Interface (UI). Any action taken users is logged into Logging and Bookkeeping (LB) as well as the status of his jobs. The VO Membership Service (VOMS) manages membership of users subscribed to a Virtual Organization (VO). The Resource Broker (RB) distributes the computational workload among the sites. The Computing Element (CE) and the Storage Element (SE) are, respectively, computational and storage resources for the Grid. The Information System (IS) collects information on the status of both Grid resources and services for resource discovering and monitoring. The LHC File Catalogue (LFC) maintains the list of Grid files strewn on the SEs and data movement services. The Grid services involved in the Grid2Win development are User Interface and Computing Element. With a graphical user interface Windows users can access the Grid graphically and with a Torque/MAUI based Computing Element MS-Windows applications can run on Grid.

A. More on Usage Interface

The User Interface (UI) is the access point to Grid facilities, it provides a set of command line tools that let users interact with most of the components of gLite in a secure, reliable and transparent way.

UI hides all the complexity of the architecture of gLite staying beyond it. You can easily install it everywhere on Internet. Many new tools are currently under development in order to more friendly Grid access. INFNGrid has developed a web-based portal called Genius [11]. There some others

cross platform solutions like GANGA [12], a GridPP project, but they are only for User Interface, Grid2Win is a native gLite implementation rebuilt under Windows. There are no projects other then Grid2win supporting a gLite windows farm. In "porting strategy" section Grid2Win Graphical User Interface will be introduced.

B. The Workload Management System

The gLite Workload Management System (WMS) is the one composed by the User Interface, the Resource Broker, the Logging and Bookkeeping and the Computing Element. The WMS is the system to submit jobs to the Grid to use its computational power. On the UI, a job is described and submitted to the RB. The description of the job is made through a Job Description Language (JDL) [13] file in which the user specifies the executable to run, the list of input files to submit together with the job, the list of files he wants to retrieve when the job terminates, requirements to execute the job and so on. Now the RB evaluates the job submission request and decides on which CE the job is sent for the actual execution. The CE first queues and then executes the job. When the job terminates, the CE notifies the result back to the RB and attaches the output sandbox. By polling the LB, the user realizes that his/her job has terminated, the output can be downloaded and the job life cycle terminates.

C. More on CE

The Computing Element is the component of gLite that provides for computational power to the Grid. Figure 2 depicts the architecture within a CE. The slashed area is the internal of the CE. The Local Resource Management System (LRMS) is a local scheduling system not necessarily Grid compliant (independent manufacturers). Local users can submit their jobs to the LRMS to be scheduled and dispatched across the Worker Nodes (WNs).

Actual LRMS supported in gLite are the Torque/MAUI, which is a for free release of the Portable Batch System (PBS) [14], Condor[15]

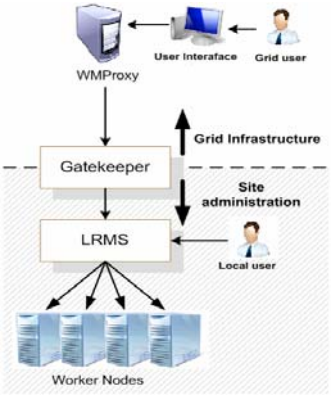


Figure 2. CE Architecture

and the Local Sharing Facility (LSF) [16]. To make the computational power of the WNs available to Grid users, the CE deploys a daemon, Gatekeeper, which is part of the Globus Toolkit. The Gatekeeper accepts submission requests from the outside Grid environment, wraps these requests into the local LRMS compliant jobs and submits them to the LRMS submission service to be scheduled. Grid users have to be mapped onto local users to own their jobs, because the LRMS is a local scheduling system. Opportune services, such are LCAS and LCMAPS, make this mapping following rules defined by the CE administrator. To execute a Grid job over the LRMS, a proper environment must be created on the WNs. To actually run, the job must first download the input sandbox from the RB, run its elaboration and then upload the output back to the RB. Moreover, on worker nodes jobs have to find a Grid compliant environment to be able to access Grid facilities.

III. PORTING STRATEGY

At the beginning of the project, we thought the best way to achieve our goals was to rebuild all the gLite services in POSIX compliant environment running on Windows.

We decided to use Cygwin as POSIX environment before rebuilding gLite source code. Cygwin [17] actually consists in a Dynamic Load Library (DLL) that converts POSIX system calls in windows systems at runtime reducing the porting issues in a large fashion. Porting of gLite took a great amount of time because of many problems during the process: first of all, not a unique building system has been adopted for the different packages. A number of not portable issues went unnoticed as gLite has been expressly built for Linux systems, from restrictions to the names of the files to issues related to the start up of services. More long time was spent on the known problem of the dynamic library loader. Since gLite Middleware binaries consist in a large number of dynamic libraries strictly depending on each other in a pyramidal way, a binary loads the first layer of libraries at run time, which in turn need to load many other libraries. The Linux library loader is able to resolve and load unresolved symbols from the library path at run time. Windows library loader is not able to do the same. For this reason we were forced to patch every Makefile to include a compile directive forcing the Cygwin gcc (the windows gcc compiler) not to leave unresolved symbols, and to handwrite all the missing library references in order to let the compiler know what library to load at compile time. Since the gLite middleware consist in more the five hundreds packages, with a variable amount of libraries in each package, and this took a rather long time. This is true especially when we consider that some packages did not use GNU auto-tool building system, but provided some makefiles only. Thus, we had to rewrite the auto-tool building system. Another point in which we got some troubles was the missing compatibility in file permission mapping. The NTFS file system version 5.0 or more is required as well as the Professional/Home Edition of Windows XP or Windows Vista.

A. Porting the User Interface: the Grid2Win GUI

The Grid2Win GUI is a Graphical User Interface running on Microsoft Windows operating systems. This GUI gives the possibility to use the Grid with the ease of any other Windows program, allowing to download/upload file with just a click (figure 3), wizard driven authentication process (figure 4), submitting a job (figure 5), wizard driven JDL creation (figure 6), graphical LDAP query (figure 7), etc.

Since this GUI is not linked to the underlying UI, it will work even if the command line programs changes (provided that their correspondent binaries still exist and the interfaces are still the same which will likely be).

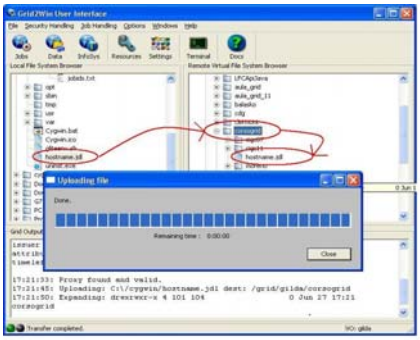


Figure 3. Data Management

Figure 4. Wizard-driven authentication process

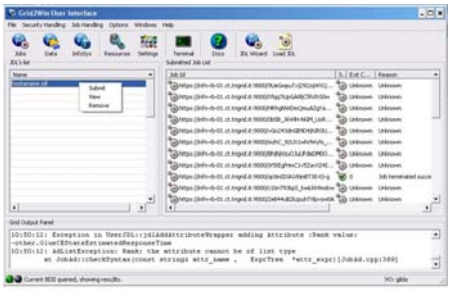


Figure 5. Job Submission.

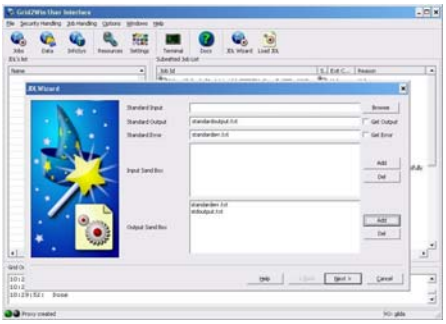


Figure 6. Wizard-driven JDL creation

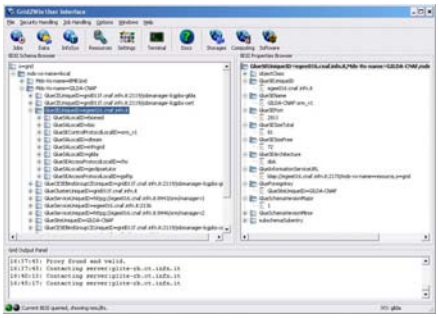


Figure 7. Graphical LDAP Query

B. Porting the Computing Element: Torque/MAUI

The next step in porting gLite on Windows was to allow computational nodes to run Windows based applications. To achieve this goal, we had to rebuild a few more tools, starting from client binaries of the Logging and Bookkeeper server to signal job status, to the globus-gass-cache file transfer clients. The first type of CE we ported to Windows was the one based on Torque/MAUI scheduler.

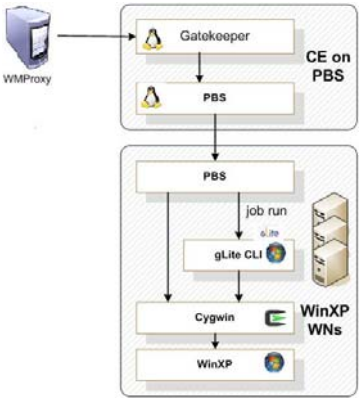


Figure 8. Torque/Maui CE implementation

In figure 8 is showed the CE; it is composed by a unique central node with Linux operating system in which the Gatekeeper and the Torque/MAUI head node are running. A set of Windows worker nodes with our rebuilt version of the gLite WN package compose the farm. Currently, one Torque/MAUI Windows based CE like the one above is deployed onto our GILDA testbed at the University of Catania and is visible at the GridICE site [18].

C. Integrating CCS into gLite

CCS is a Microsoft Windows Server 2003 x64 native scheduling system composed by a head node, in charge of accepting job submission requests, and a farm of worker nodes. The head node provides for canonical job submission primitives through a command line interface as well shell commands. To integrate the Compute Cluster Server (CCS) into gLite, we had to define a temporary architecture shown in Figure 9.

As shown in Fig. 9, the architecture is composed by the CCS Windows farm and an Linux auxiliary node. The need for the latter one is due to the fact that the Globus Gatekeeper executes under Linux.

To put in communication the auxiliary node with the CCS head node, we coded a novel

service of ours (the wxCCSJob) that works over a secure channel and supports data movement.

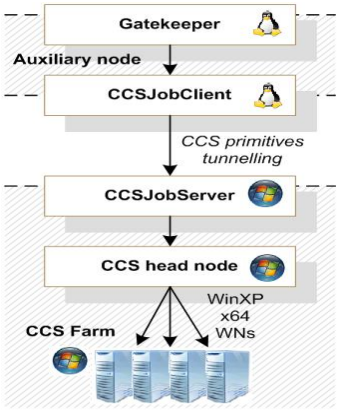


Figure 9 - CSS Windows farm

Aim of the wxCCSJob is to allow the Gatekeeper to call CCS head node primitives even if they run on different machines and under different operating systems. However, this is a temporary solution to be replaced by the adoption of Windows version of the Gatekeeper that we are going to rebuild.

IV. Conclusion

In this paper we have discussed about Grid2Win project and reported results already achieved. You can consult the project home page [19] for more details and user guide: system requirements and first installation. We detailed more on CE architecture and techniques to port the code without the need to rewrite any service. While still in Alpha or Beta stage, we successfully achieved to run a parallel and well integrated Windows-compatible Grid on the existing GILDA infrastructure with the gLite middleware, and, on some conditions (PBS local scheduler) we are even in production with a cluster of Windows Worker Nodes in our site in Catania. In the next future we are planning to make some changes. First of all we are going to migrate to a Microsoft native POSIX environment instead to improve performances of the services. Furthermore, we plan to remove the auxiliary node from the CCS based CE by porting the Gatekeeper and related services to Windows. We want to create plug and play Worker Nodes based on the seti@home [20] paradigm allowing to use idle PCs computing power during their idle time. Last, we are going to integrate our build system with gLite build System and merge windows patched/portability issues to the gLite Official CVS.

References

[1] I. Foster and C. Kesselman. "The Grid: blueprint for a new Computing Infrastructure", Book, Ed. Morkan Kauffman, 1997
[2] I. Foster, C. Kesselman and S. Tueckle, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", in Lecture Notes in Computer Science, Vol. 2150, 2001.
[3] I. Foster, "The Grid: A New Infrastructure for 21st Century Science", in Physics Today, Vol. 55, pp. 42-27, 2002.
[4] LHC Experiment home page: public.web.cern.ch/public/en/LHC/LHCExperiments-en.html
[5] INFN – Catania home page: www.ct.infn.it/
[6] gLite project homepage, www.glite.org/
[7] EGEE project home page: www.eu-egee.org/ .
[8] Compute Cluster Server project homepage, www.microsoft.com/windowsserver2003/ccs/default.mspx
[9] I. Foster and C. Kesselman, "Globus: A Metacomputing Infrastructure Toolkit", in Internation Journal of Supercomputer Applications, Vol. 11, pp-115-128, 1997.
[10] Globus Toolkit (GT) homepage, www.globus.org
[11] Genius portal home page: glite-tutor.ct.infn.it/
[12] GANGA project home page: ganga.web.cern.ch/ganga/
[13] Job Description Languange documentation: edms.cern.ch/file/590869/1/EGEE-JRA1-TEC-590869-JDL-Attributes-v0-9.pdf
[14] Open Portable Batch System (PBS) homepage, www.openpbs.org/about.html
[15] Condor project homepage, www.cs.wisc.edu/condor/
[16] Platform homepage, www.platform.com/
[17] Cygwin home page: www.cygwin.com/
[18] GridICE,alifarm7.ct.infn.it:50080/gridice/site/site.php
[19] Grid2Win home page: grid2win.forge.eu-eela.eu/new/index.php
[20] Seti@Home project homepage, setiathome.ssl.berkeley.edu/

The GENIUS Grid Portal and Robot Certificates

Roberto Barbera[1,2], Giacinto Donvito[3], Alberto Falzone[4],Giuseppe La Rocca[1], Luciano Milanesi[5], Giorgio Pietro Maggi[3]

[1]*INFN Catania – Via S. Sofia 64, I-95123 Catania, Italy {roberto.barbera,giuseppe.larocca}@ct.infn.it*
[2]*University of Catania – Via A. Doria, 6, I-95125 Catania, Italy, roberto.barbera@ct.infn.i*
[3]*INFN Bari – Via E. Orabona 4, I-70126 Bari, Italy, {giacinto.donvito,giorgio.maggi}@ba.infn.it*
[4]*NICE Srl – Via Marchesi di Roero 1, I-14020 Cortanze (AT), Italy, alberto.falzone@nice-italy.com*
[5]*Instituto di Tecnologie Biomediche – CNR – Via F. Cervi 93, I-20090 Segrate(MI), Italy, luciano.milanesi@itb.cnr.it*

Abstract— Grid Portals are increasingly used to provide user friendly interfaces for users interested to access distributed Grid resources. In this paper the new solution implemented into the GENIUS Grid portal to allow users, not holding a personal X.509 certificate and therefore not belonging to a specific Virtual Organization, to access the grid resources by means of a robot certificate will be shown. This new solution strongly simplify the Grid access of new potential users and represents a valuable step forward to speed up the adoption of grid paradigm amongst new scientific communities.

Index Te1ms— Grid Computing, Grid portals, robot certificates.

I. INTRODUCTION

THE massive use of e-Infrastructures is rapidly changing the landscape of science. Remote access to computing services, instrumentation and resources in general, creates new opportunities for researchers to bring existing applications to higher levels of usability and performance. Modern e-Infrastructures enable researchers to deploy new strategies in approaching scientific problems and can contribute to stimulate the creation of new scientific communities, joining researchers who are working on similar challenges and are willing to share resources and reach new levels of collaboration. Researchers can gain access to scientific data and unique instruments located in top level laboratories around the world without the need to travel to speed up the creation of results of scientific relevance and open new unthinkable scenarios up to few years ago. This is why these modern ICT-based infrastructure are going to be spread into different scientific domains such us: Astronomy, Computational Chemistry, Earth Science, Financial Simulation, High Energy Physics and Biomedicine as well. An important component of a e-Infrastructure is represented by the Grid Infrastructure, a revolutionary distributed environment for sharing heterogeneous computing resources and mass storage system

distributed world wide and interconnected by large bandwidth networks.
Unfortunately, up to now, the basic know-how requested to access e-Infrastructures is not so trivial, especially for not ITC expert users. There is a sort of *"scientific gap"*, represented by the difficult of complex computing protocols and rather complex Command Line Interface that modern scientists have to learn before they access Grids and gain their benefits. Surely, the high security policy requested to access the distributed computing resources is a rather big limiting factor to increase the usage of Grids to a wide community of users. Grid security is indeed based on the Public Key Infrastructure of X.509 certificates and the procedure to get and manage these certificates is unfortunately not straightforward. Using the Web technology and its new recent developments, Grid details can be hidden to the end users, giving access to the infrastructures in a very easy way as such as the common sense in web usage. GENIUS Grid Portal[1][2], powered by EnginFrame[3], is a computing web portal which provides to the end-user a transparent and abstraction layer to access the distributed services and resources of a typical grid infrastructure. This is why it's considered a *grid gateway*.

In this paper we are going to describe the work performed on grid with robot certificates and the GENIUS Grid Portal. The paper is organized into sections. In the first one robot certificate and its main features will be introduced, then will be presented the GENIUS Grid Portal and described how its architecture has been extended to support robot certificates. Finally, we describe the set up of a distributed environment that allows non-expert users to exploit Grid resources for large-scale phylogenetic analysis.

A. The Robot Certificates

Starting from Feb. 2008 the Italian INFN Certification Authority (CA) modified its CP/CPS *(Certification Policy and Certification Practice Statement of a Certification Authority)* to permit users to apply for robot certificates which now are officially recognized as a standard by the IGTF[4]. Robot certificates (also known as *portal certificate*) have recently been introduced to perform automated tasks on the Grid on behalf of users. They are extremely useful for example to automate grid service monitoring, data processing production, distributed data collection systems and others. Since all these tasks are basically unattended and not necessarily located on a specific host, the decryption of the private key with a passphrase is not feasible. Basically robot certificates can be used to identify a person responsible for an unattended service or process acting as client and/or server. These certificates may be used to authenticate the service to another Grid entity, possibly by signing proxy certificates. The overall characteristic of a robot certificate is that its includes the full name of the subscriber, as determined by the Virtual Organization/Site's RA, as well as a label reflecting the purpose of the robot entity (`e.g. /C=IT/O=GILDA/L=INFN Catania/CN=Robot: MrBayes – Giuseppe La Rocca`). In this example MrBayes is the function related to this certificate. All the users belonging to the GILDA VO will use this robot certificate to run this specific application. Fig. 1 shows the (Distinguished Name) DN of a typical robot certificate.

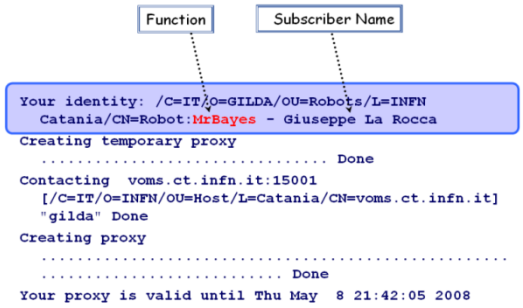


Fig. 1 – The DN of a typical robot certificate

In order to strong reduce the risk to have the portal certificate lost or compromised, the

INFN CA decided to issue this new certificate on the *Aladdin eToken[5]* smart card. Each smart card can support several robot certificates: one for each application we want to share with other users of the same VO. An user's PIN is prompted every time the certificate stored on the smart card is read to generate a proxy. This additional feature helps to keep the certificate protected and safe. Tab. 1 shows the technical specification of this hardware.

Operating systems	Windows 98/98SE/Me/2000/XP/NT4.0 SP6 and later/Vista; Mac OS X; Linux
API & standards support	PKCS#11 v2.01, CAPI (Microsoft Crypto API), Siemens/Infineon APDU commands PC/SC X.509 v3 certificate storage, SSL v3, IPSec/IKE
Models (by memory size)	32K, 64K
On board security algorithms	RSA 1024-bit / 2048-bit, DES, 3DES (Triple DES), SHA1
Security Certifications	Common Criteria EAL5/EAL5 + (smart card chip) /EAL4 + (smart card OS)
ISO specification support	Support for ISO 7816 1 to 4 specifications
Memory card retention	At least 10 years
Memory call rewrites	At least 500,000

Tab. 1 - eToken PRO Smartcard specification

Once the robot certificate is safely stored on the eToken smart card, the `mkproxy` shell script developed at NIKHEF[6] can be used to generate grid proxies reading the certificate directly from it. This script requires some external libraries and software, which need to be installed before attempting to use it. The script works perfectly on : Windows XP (using cygwin), Fedora Core 5, Linux CentOS 4, Scientific Linux 4, Linux OpenSuse 10. In the near future some tests will be performed on MacOS X as well. Starting from a single robot certificate installed on board of the Aladdin eToken PRO, user can generate a plain grid proxy by issuing the command as follows:

```
mkproxy –-label="Robot:MrBayes"

Starting Aladdin eToken PRO proxy
generation
Found X.509 certificate on eToken:
label: (eTCAPI) Robot:MrBayes –
Giuseppe La Rocca's GILDA ID
id:
39453945373335312d333545442d343031612d3
84637302d32384636363930363042303a31
Your identity: /C=IT/O=GILDA/L=INFN
Catania/CN=Robot:MrBayes – Giuseppe La
Rocca
Generating a 512 bit RSA private key
..........++++++++++++
......++++++++++++
writing new private key to
'proxykey.D17633'
engine "pkcs11" set.
Signature ok
subject=/C=IT/O=GILDA/L=INFN
Catania/CN=Robot:MrBayes – Giuseppe La
Rocca /CN=proxy
Getting CA Private Key
PKCS#11 token PIN:
Your proxy is valid until:
Sun Feb 24 03:58:09 CEST 2008-02-23
```

In this example a plain proxy is signed by the robot certificate stored on the eToken with the given label. Once the plain grid proxy is generated, it's possible to add the VOMS (Virtual Organization Management Service) extensions by issuing the command `voms-proxy-init` using the `noregen` option.

B. The GENIUS Architecture and EnginFrame

EnginFrame is a Web-based technology, developed by the Italian company Nice[7] S.r.l., that enables the access and the exploitation of Grid-enabled applications and infrastructure. It allows users, grouped within of a VO, to exploit the whole power of a grid infrastructure hiding all its complexity to the end-users and providing an abstraction layer to the grid technologies. *EnginFrame* is also well known in the Grid research world for being the technology which *GENIUS (Grid Enable web eNvironment for site Independent User job Submission*), the official Grid portal of the European project EGEE is based on. Due to the modularity and flexibility of EnginFrame, the Genius Portal can be easily customized to interact with several Grid component and/or VO allowing scientists to access, execute and monitor their own applications only by using a simple web browser. In this way the complexity of the underlying Grid

infrastructure is hidden to users. In this work the EnginFrame framework has been extended (see fig.2) to support these new certificates and simplify the access to Grids of non-expert users.

Fig. 2 – EnginFrame framework enhanced to support robot certificates

From now on when the smart card is inserted in the server where GENIUS is running, the portal starts to generate a new user's proxy signed by the robot certificate stored in the usb token, otherwise the normal authentication based on a dedicated Java applet will be performed. Once the proxy is generated the user is automatically redirected to the home page of the application associated with the robot certificate. The Aladdin eToken PRO smart card is totally transparent for the end-user. If the smart card is available on the server, an automatic frame will drive the user to create a temporary proxy before to run the available application in grid.

C. Users Tracking Systems (UTS)

Since with robot certificates the access to a grid infrastructures is opened to a wide basin of users (just an account on the server is needed), in order to enhance the system security and account the payload produced by users in grid an Users Tracking System (UTS) has been deployed into the GENIUS architecture. Based on an underlying MySQL database and on PHP, JavaScript and XML code, the system allows to observe and register the tasks performed in Grid by robot certificate. The portal administrator is the only one that can later access the collected accounting data. Some dedicated services with restricted Access Control List (ACL) have been designed into the portal in order to allow the administrator to interact with the UTS using a web interface (Fig.3). By using this system it is possible to monitor the behavior of who is working on the grid resources using robot certificates.

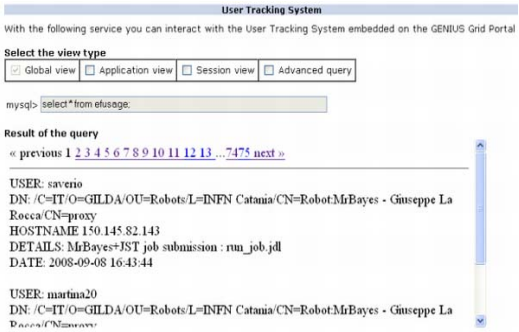


Fig. 3 – Retrieve information querying the UTS.

D. Bayesian Phylogenetic Inference on a large scale

The GENIUS Grid portal with its transparent support to the robot certificates has been successfully used by non-grid users to run a bioinformatics application in Grid. In this section some details about the application's workflow which has been set up in order to perform phylogenetic analysis on a large scale is shown.

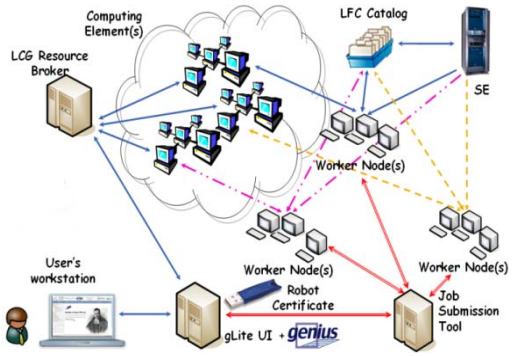


Fig. 4 – The workflow set up to perform phylogenetic analysis on a large scale

The client side is represented by a user's workstation running a web browser. The server side is represented by a gLite UI machine, equipped with the middleware services to submit jobs and manage data on Grid, the Apache Web Server, the Java/XML portal framework EnginFrame developed by NICE SRL and the GENIUS Grid Portal itself. After user's login, a proxy certificate is requested by the portal to grant access to the distributed resources of a Grid Infrastructure according to the GSI[8] standard. If the USB eToken is plugged into the server, the portal will generate the needed proxy on user's behalf using the robot certificate stored on it. This operation is completely transparent for the end-user. In a few minutes the robot certificate stored on the USB token is read to generate the proxy certificate. Once the proxy has been successfully created, the user is automatically redirected to the home page of the application closely related to this certificate. Here, through the services developed for this application, the user can provide his/her input settings before to launch the parallel computation in Grid. Moreover, in order to improve the reliability of this workflow and dealt with possible job failures a JST (Job Submission Tool)[9] has been also introduced in the architecture. This tool has been adopted for the submission of large number of jobs in an almost unattended way. It is based on the concept of "task" to be executed. The entire problem is first subdivided into elementary tasks: then all the tasks are inserted into a DB server. In the submission phase all the jobs are identical, in fact when the job is submitted it does not know which task has to execute. Only when the jobs lands and starts executing on a worker node, it requests to the central DB a task to execute. Information on the execution of each task is logged in the central DB. Only if all steps are correctly executed by the job, the status of that particular task on the central DB is updated to "Done". In this way the central DB provides a monitoring of the task execution and no manual intervention is required to manage the resubmission of the failed tasks: tasks which are found in a "running" state after a given time interval, are considered failed and automatically reassigned to new jobs. Figure 5 shows the monitoring and the visualization system developed for the bioinformatics application.

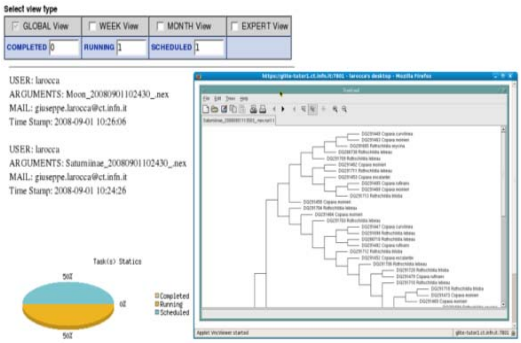


Fig. 5 - Monitoring and visualization services

II. CONCLUSION

The work carried out by INFN is particular important for all users who are not familiar or do not want to be familiar with personal digital certificates and the internals of the Grid Security Infrastructure. The benefits introduced by this work are far-reaching for several user communities and applications. This work demonstrates how robot certificates can really contribute to make Grid technology more appealing for end users providing an asset in raising Grid awareness to a larger number of potential users. GENIUS Grid Portal is the official portal of GILDA[10], a complete t-Infrastructure for grid dissemination and training set up and managed by the INFN in the context of the EGEE[11] Projects. EnginFrame has constituted a valid, secure, robust and flexible infrastructure on which we have built all of GENIUS Services used by EGEE community on various grid sites. Other regional projects as such as Trigrid[12] and PI2S2[13] ones are adopting GENIUS portal

with success, porting on the web interface many applications running on their infrastructure, being so a powerful gateway to the Grid resources with the needed required security.

ACKNOWLEDGMENTS

We gratefully acknowledge all the people who supported this work contributing with ideas, requirements and feedback. We would like to warmly thank Jan Just Keijser (janjust@nikhef.nl) from NIKHEF for his technical support.

REFERENCES

[1] Andronico G, Barbera R, Falzone A, Lo Re G, Pulvirenti A, Rodolico A **– "GENIUS: a web portal for the grid" –** Nucl. Instrument and Methods in Phy. Res. A 2003. Visit also the official GENIUS web site: https://genius.ct.infn.it/

[2] Barbera R, Falzone A, Ardizzone V, Scardaci D – "The GENIUS Grid Portal : Its Architecture, Improvements of Features ,and New Implementations about Authentication and Authorization" – Enabling Technologies: Infrastructure for Collaborative Enterprises, 2007. WETICE 2007. 16th IEEE International Workshops on.

[3] EnginFrame – www.enfingrame.com/

[4] IGTF – www.gridpma.org

[5] Aladdin eToken – www.aladdin.com/etoken/

[6] NIKHEF – www.nikhef.nl/

[7] NICE Srl – www.nice-italy.com/

[8] GSI – www.globus.org/security/overview.html

[9] JST – Job Submission tool - De Sario G, Gisel A, Tulipano A, Donvito G, Maggi G P– "**High-throughput GRID computing for Life Sciences, in Mario Cannataro (Ed.), Handbook of Research on Computational Grid Technologies for Life Sciences, Biomedicine and Healthcare, IGI Global"** (to appear) – Visit also the JST web site : webcms.ba.infn.it/cms-software/index.html/index.php/Main/JobSubmissionTool

[10] Andronico G, Ardizzone V, Barbera R, Catania R, Falzone A, Giorgio E, La Rocca G, Monforte S, Pappalardo M, Passaro G, Platania G, **– GILDA: The Grid INFN Virtual Laboratory for Dissemination Activities.** Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005 **–** Visit also the official GILDA web site : https://gilda.ct.infn.it/

[11] The EGEE Project – public.eu-egee.org/

[12] The TRIGRID Project – www.trigrid.it/

[13] The PI2S2 Project – www.pi2s2.it/

CRESCO HPC System Integrated into ENEA-GRID Environment

G. Bracco[1], S. Podda[1], S. Migliori[1], P. D'Angelo[1], A. Quintiliani[1], D. Giammattei[1], M. De Rosa[1], S. Pierattini[1], G. Furini[1], R. Guadagni[1], F. Simoni[1], A. Perozziello[1], A. De Gaetano[1], S. Pecoraro[1], A. Santoro[1], C. Sció[2], A. Rocchi[1], A. Funel[1], S. Raia[1], G. Aprea[1], U. Ferrara[1], D. Novi[3] and G. Guarnieri[1]

[1]*ENEA-FIM, C.R. ENEA Via E. Fermi 45, 00044 Frascati (Rome) Italy*
bracco@enea.it
[2]*Esse3Esse, Rome, Italy*
[3]*CRIAI, Portici (Napoli), Italy*

***Abstract*—This paper describes the architecture of the High Performance Computing (HPC) system that has been installed to provide the required computing power to the CRESCO project applications and the dedicated activity required to integrate the HPC system into ENEA-GRID infrastructure. ENEA, the Italian agency for the energy, environment and new technologies, has a substantial experience in GRID technologies and its multi-platform HPC resources are integrated in the ENEA-GRID infrastructure. CRESCO (Computational Research Center for Complex Systems) is an ENEA Project, co-funded by the Italian Ministry of University and Research (MUR). The project is functionally built around a HPC platform and 3 scientific thematic laboratories: (a) the Computing Science Laboratory (b) the Life Science domain Laboratory and (c) the Complex Network System Laboratory. CRESCO HPC system consists of more then 2700 computing cores, divided into three main sections. A section is dedicated to applications requiring large memory, a section dedicated to high scalable applications and a third experimental section providing systems with Cell processors and high performance video adapters dedicated to computational applications. High bandwidth and low latency connections are provided by an InfiniBand network. The main storage is organized in a GPFS file system. CRESCO HPC system has been integrated into ENEA-GRID infrastructure which has been developed to provide a unified environment for all the main ENEA HPC resources. The main software components of ENEA-GRID are the multi-site resource manager LSF Multicluster, the OpenAFS distributed file system, the integrated Kerberos 5 authentication and a Java and Web based Graphical User Interface making use of CITRIX technologies. The choice of mature, reliable and multi-platform software components has permitted along the years to integrate in a GRID oriented infrastructure HPC resources at the state of the art performances, with minimal changes in the user environment.**

***Index Terms*—ENEA-GRID, HPC, CRESCO, GRID Computing.**

I. Introduction

ENEA, the Italian agency for the energy, environment and new technologies, has a substantial experience [1] in GRID technologies and its multi-platform HPC resources are integrated in the ENEA-GRID infrastructure [2]. This paper describes the architecture of the High Performance Computing (HPC) system that has been installed to provide the required computing power to the CRESCO project applications and the dedicated activity required to integrate CRESCO HPC system into ENEA-GRID infrastructure. CRESCO (Computational Research Center for Complex Systems) [3] is an ENEA Project, co-funded by the Italian Ministry of University and Research (MUR). The project is functionally built around a HPC platform and 3 scientific thematic laboratories:

(a) the Computing Science Laboratory, hosting activities on Hardware and Software design, GRID technology and HPC platform management; (b) the Computational Systems Biology Laboratory, with activities in the Life Science domain, ranging from the post-omic sciences (genomics, interactomics, metabolomics) to Systems Biology; (c) the Complex Networks Systems Laboratory, hosting activities on complex technological infrastructures, for the analysis of Large National Critical Infrastructures.

A. CRESCO HPC System

CRESCO HPC system has been designed with the aim of offering a general purpose facility based on the leading multi-core x86_64 technology. The performance for the CRESCO HPC system [4] has ranked 125 in the June 2008 top500 list [5] with Rmax=17.1 TeraFlops in the HPL benchmark (rank 2 between the Italian HPC systems in the list). In order to provide the best environment for different types of applications the system consists of two main sections respectively oriented (1) for high memory request and moderate parallel scalability and (2) for limited memory and high scalability cases. Both sections are interconnected by a common InfiniBand 4X DDR network (IB) and can operate as a single large integrated system.

1) CRESCO Section 1, for Large memory application: The first main section is composed by 42 fat nodes IBM x3850-M2 with 4 Xeon Quad-Core Tigerton E7330 processors (2.4GHz, 1066MHz, 6MB L2), 32 GB RAM (12 extra-fat nodes with 64 GB RAM). The total number of cores in the first section is then equal to 672.

2) CRESCO Section 2, for high scalable applications: The second main section is composed by 256 blades IBM HS21 each supporting dual Xeon Quad-Core Clovertown E5345 processors (2.33GHz, 1333MHz, 8MB L2), 16 GB RAM for total of 2048 cores. The blades are hosted by the 14 slots blade chassis for a total of 19 chassis and each blade has a dedicated IB connection. The larger system created by joining the two main sections has 2720 cores.

3) CRESCO Section 3, experimental resources: A third experimental section consists of 3 subsections dedicated to special processor architectures: 4 blades IBM QS21 with 2 Cell BE Processors 3.2 GHz; 6 nodes IBM x3755, 4 sockets AMD Dualcore 8222 equipped with a FPGA VIRTEX5 LX330 card 4 node IBM x3755, 4 sockets AMD Dualcore 8222 with a NVIDIA Quadro FX 4500X2 (3 machines) and 4700X2 (1 machine) video card; the latter support CUDA while all of them support the GP-GPU development environment.

4) CRESCO network: The IB network (4X DDR) is based on a set of CISCO IB switches, model SFS 7024 (288 ports), model SFS 7012 (144 ports) and 5 model SFS 7000 (24 ports for a total of 120n ports). A schematic of CRESCO IB network is shown in Fig.1. The GigaEthernet network consists of one switch CISCO 4506 (240 ports), 3 switches CISCO 4948 (144 ports) and 3 switches CISCO 3750G (144 ports) providing both the basic network connection, the management LAN and a dedicated GigaEthernet network for parallel applications.

5) CRESCO storage and backup: The storage of CRESCO HPC system is provided by an IBM DCS9550 system, 160 TB raw space, based on 500 GB SATA Hard Disks; the data space is organized in a GPFS file system, with 4 NSD server connected to the storage by FC interfaces. The GPFS parallel file system is shared via InfiniBand between all the computing nodes of all the main section of the system. Two IBM DS4200 and DS3400 storages devices provide the AFS data space. An IBM Tape Library IBM TS3500 provides the backup facility.

6) CRESCO software: The operating system is RedHat EL 5.1 and the usual set of GNU, Portland and Intel Compilers are available together with the standard OFED [6] MPI stacks (openmpi, mvapich, mvapich2). User homes are located in the OpenAFS file system, one of the base elements of the ENEA-GRID infrastructure.

7) CRESCO installation: The three sections together with more than 70 service machines (front-end, controls, file-servers, instal-

Fig. 1. Schematic of the CRESCO InfiniBand (4X DDR) Network; the schema includes also the FC connection from the GPFS NSD servers to the DDN storage controllers

lation servers, GRID interface machines,...) and storage and network components make use of a total of 20 standard racks (19 inch, 42 U). The power required to run the system has been estimated to 150 kw and a proper cooling system has been designed and installed. CRESCO computer room is shown in Fig.2.

Fig. 2. CRESCO Computer Room

B. ENEA-GRID infrastructure

In the last decade ENEA has tried to ride the wave of the information technology explosion by the procurement of computational systems at the leading edge of the available technology, taking care of the development of a stable software infrastructure capable to provide ENEA researchers and their collaborators of an friendly and consistent access to the available resources [7]. In this framework GRID technologies have been adopted and powerful HPC (High Performance Computing) systems have been acquired, taking advantage also of the participation in national programs from the Italian Ministry of University and Research. The exploitation of these hardware resources follows the requests of ENEA researchers, that cover a wide spectra of applications in the domains of simulation, modelling, data analysis and 3D visualization. In this context the choice of the computational system architecture has been very diversified over the years, so that ENEA has now a wide pool of different architectures and operating system (AIX, Linux, IRIX, MacOSX, Solaris) accessible through an integrated GRID-based software platform called ENEA-GRID CRESCO HPC system, located in Portici (Naples) ENEA research center, has been integrated into ENEA-GRID and at the moment is the most relevant computational resource of the infrastructure.

1) ENEA-GRID Architecture: The GRID approach has permitted to integrate in a unified system all the high performance computational resources available in ENEA, distributed in many of its 12 research sites located in northern, central and southern Italy, see Fig.3. Local computational facilities are managed in the computer centers in each of the main sites and include resources for serial and parallel computing and advanced 2D/3D graphical post processing. ENEA-GRID infrastructure permits to access to all these resources as a single virtual system provided by multiplatform systems (AIX/IBM SP, Linux x86, x86_64, IA64, IRIX/SGI, MacOSX).

2) ENEA-GRID components: ENEA-GRID main components are (1) a GRID enabled resource manager, LSF (Load Sharing Facility) in its Multicluster version [8] and (2) a distributed file system, AFS (Andrew File System), presently an open source project with the name OpenAFS [9] and (3) a unified graphical user interface (based on Java and Citrix technologies [10]). All these components constitute the so-called Middleware of the ENEAGRID infrastructure. ENEA GRID architecture has been based on mature middleware components, both proprietary and open sources, in order to provide good performances and service relia-

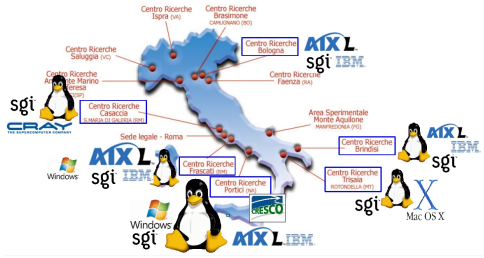


Fig. 3. Location of ENEA-GRID resources in Italy

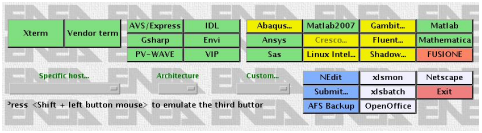


Fig. 4. ENEA-GRID Graphical User Interface

bility so that a production level quality can be attained. This choice has permitted along the years to integrate in ENEAGRID infrastructure HPC resources at the state of the art performances, with minimal changes in the user environment. The ENEA-GRID graphical user interface, see Fig.4 has been integrated with a specific GUI dedicated to the access to CRESCO HPC resources, see Fig.5. Recently a freenx [11] based desktop access has also been installed in CRESCO, providing an effective open source solution for the remote graphical user access, see Fig.6. The management and the user services available in ENEA-GRID has been recently improved by a set of new utilities and tools, described in reference [13].

3) ENEAGRID network connection: The ENEA-GRID sites are connected by GARR, the Italian Academic and Research Network, see Fig. 7. The connection bandwith provided by the GARR Network ranges from 2 Gbps of the Portici site, where CRESCO HPC system is located, to 1 Gbps for the Frascati site, hosting the main AIX/SP4/SP5 resources. Other sites are connected at lower bandwidth values (Casaccia, 200 Mbps, Brindisi 150 Mbps, Trisaia 18 Mbps, Bologna 30 Mbps)

4) ENEAGRID interoperability with other GRID infrastructure: ENEA has participated to

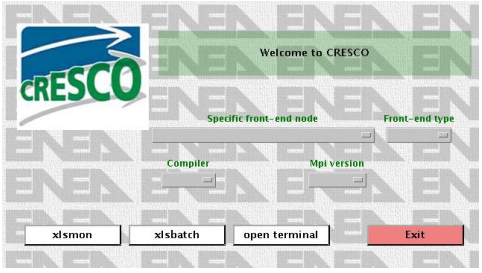
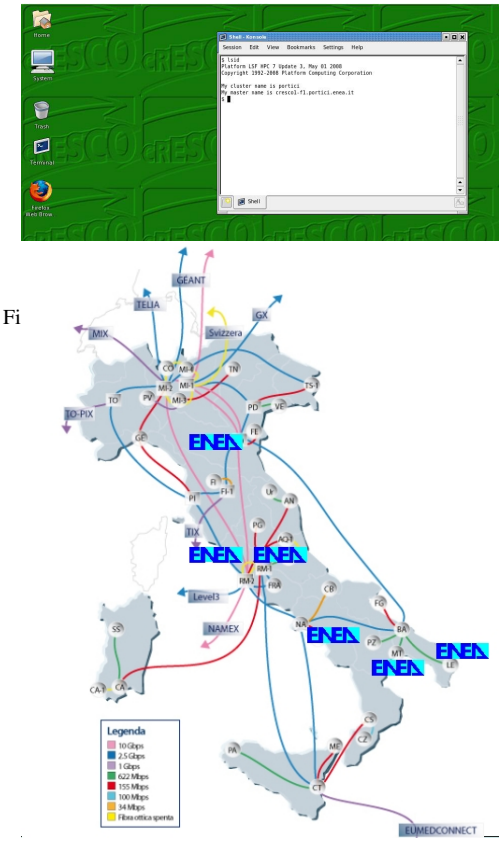


Fig. 5. ENEA-GRID/CRESCO Graphical User Interface

many GRID projects focusing the participation in the interoperability with the other existing GRID infrastructures. To implement the interoperability ENEA has developed an original solution based on the SPAGO concept (Shared Proxy Approach for GRID Objects), see reference [12] in this conference. The utilization of SPAGO based gateway has enabled the interoperation between CRESCO HPC center and the GRID infrastructure born to include all the main computation resources available in the Southern



Fi

Fig. 7. GARR Network and ENEA main computer sites

Italy area, integrated in the GRISU activity [14].

II. Conclusion

CRESCO HPC system provides the computational resources for the applications of the thematic area of the CRESCO project. CRESCO HPC is one of the leading Italian HPC facility and is fully integrated into the ENEA-GRID infrastructure, so that it can be used by ENEA researchers and their collaborators through the standard ENEA-GRID tools and interfaces, minimizing in this way the user effort for the access to the new computational facility.

References

[1] S. Migliori, et al. ENEA Computing Factory, Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 1999, June 28 July 1, 1999, Las Vegas, Nevada, USA. Vol.6, 3037-3040
[2] http://www.eneagrid.enea.it.
[3] http://www.cresco.enea.it.
[4] International Supercomputing Conference, Dresden 17-19/6/2008 "Architecture and performances of the CRESCO HPC system" S. Migliori, G. Bracco, P. D'Angelo, D. Giammattei, M. De Rosa, S. Pierattini, S. Podda, A. Quintiliani,S. Raia,A. Funel, G.Richelli, Research Poster#7
[5] http://www.top500.org.
[6] http://www.openfabrics.org.
[7] Conferenza GARR 2005 "La rete daPERtutto", Pisa, 10 - 13/5/2005 "Larchitettura di ENEA-GRID, infrastruttura distribuita per la condivisione di risorse informatiche per il calcolo scientifico".
[8] http://www.platform.com.
[9] http://www.openafs.org.
[10] http://www.citrix.com.
[11] http://freenx.berlios.de
[12] "ENEA-GRID and glite Interoperability: robustness of SPAGO approach" A. Santoro, C.Sció, G. Bracco, S. Migliori, S. Podda, A. Quintiliani, This conference Paper #43
[13] "New ENEA-GRID Integrated Support Services for CRESCO" A. Rocchi, A. Santoro, C.Sció, G. Bracco, S. Migliori, S. Podda, A. Quintiliani, This conference Paper #49
[14] http://www.grisu-org.it/

G. Bracco graduated in Physics at Pavia University (1976) and after a short experience with IBM Italia started research activity in the Nuclear Fusion domain, at CEA Fontenay-aux-Roses (France) and then (1981) at ENEA Frascati Laboratory (Italy). In 2003 he joined the ENEA computer division, working on GRID and High Performance Computing.

ENEA-GRID and gLite Interoperability: Robustness of SPAGO Approach

Andrea Santoro[1], Carlo Sciò [2], Alessio Rocchi[1], Giovanni Bracco[1], Silvio Migliori[1], Andrea Quintiliani[1], Salvatore Podda[1]

[1]*C.R. ENEA Frascati, Frascati, Italy, andrea.santoro@frascati.enea.it*
[2]*Esse3Esse, Roma, Italy, scio@frascati.enea.it*

Abstract—**The GRID approach has allowed to integrate in a single unified system, ENEA-GRID, all the high performance computational resources available in ENEA. In the participation of ENEA in national and European GRID projects, solutions had to be found to enable the interoperability between ENEA-GRID and the gLite middleware. The ongoing activity to make ENEA-GRID interoperable with the EGEE Grid infrastructure is presented here. The interoperability has been based on the SPAGO technology (Shared Proxy Approach for Grid Objects) which solves the issues coming from the different mechanisms for authentication, scheduling and data sharing and the ENEA EGEE site is currently in production as previously documented. However, the recent platform transition of the gLite infrastructure (from SL3 to SL4) has required reworking some components of the SPAGO architecture. Nonetheless the overall SPAGO approach has been basically untouched and we have been able to reapply it after a few modifications, proving its robustness to technology changes. The same solutions have also been adopted in implementing the interoperability with another gLite based infrastructure, namely the Grid connecting the computational resources of four projects in Southern Italy (CRESCO, SCOPE, CYBERSAR and PI2S2), which is now interfaced using gLite 3.1.**

***Index Te1ms*— SPAGO, interoperability, GRISU, gLite, ENEA-GRID.**

I. INTRODUCTION

The Grid approach has allowed to integrate in a single unified system, ENEA-GRID, all the high performance computational resources available in ENEA. The main software components of ENEA-GRID are the distributed file system OpenAFS[7] and the multi-site resource manager LSF Multicluster[8], which constitute the production ready ENEA-GRID middleware. In the participation of ENEA in national and European GRID projects, solutions had to be found to enable the interoperability between ENEA-GRID and other Grid middlewares. Among these gLite [3] is one of the leading solutions, originated in the EGEE project and adopted in other contexts. The ongoing activity to make ENEA-GRID interoperable with the EGEE Grid infrastructure is presented here. The interoperability has been implemented according to a gateway approach. This approach heavily relies on the SPAGO technology (Shared Proxy Approach for Grid Objects) which solves the issues coming from the different mechanisms for authentication, scheduling and data sharing. The ENEA EGEE site is currently in production using this technology [2,9,10]. However, the recent platform transition of the gLite infrastructure (from SL3 to SL4) has required reworking some components of the SPAGO architecture. Also, in the transition from gLite 3.0 to gLite 3.1, the native AFS filesystem support has

been discontinued. Nonetheless the overall SPAGO approach has been basically untouched and we have been able to reapply it after few modifications, proving its robustness to technology changes. The same solutions have also been adopted in implementing the interoperability with another gLite based infrastructure, namely the Grid connecting the computational resources of four projects in Southern Italy (CRESCO, SCOPE, CYBERSAR and PI2S2), which is now interfaced using gLite 3.1. CRESCO computation resources are fully integrated into ENEA-GRID and the solutions found for EGEE have been straightforwardly applied, in a context where each of the 4 projects retains the capability to run all the main Grid services. The rest of this paper is organized as follows: section II details the gateway approach, in its two components, including the SPAGO technology; section III details the challenges to overcome in the gLite transition from SL3 to SL4; section IV describes how the gateway approach allowed us to make CRESCO interoperable with the other PON1575 projects.

II. Gateway Approach

The gLite software is the Grid middleware developed in the context of the EGEE project. Its authentication scheme is based on X509 certificates as an authentication mechanism, its data sharing is enabled through the existence of Storage Elements visible to the whole Grid, and the scheduling is carried out by a software element known as Workload Management System (WMS). Conversely, ENEA-GRID employs a middleware based on the geographically distributed filesystem OpenAFS for data sharing, the resource manager LSF Multicluster for scheduling, and a combination of Kerberos[6] and AFS tokens as authentication mechanism. Ensuring interoperability between such different systems consists of two different parts. Submitting jobs from ENEA-GRID to gLite and submitting jobs from gLite to ENEA-GRID. We will look at them separately.

A. ENEA-GRID to gLite

Job submission from ENEA-GRID to gLite is made difficult by the fact that the submission interfaces of the two middlewares are very different. Specifically, LSF multicluster does not have embedded supports to interface with gLite middleware. Therefore, the overall design approach, shown in Figure 1, is the following: an ENEA-GRID user who wants to submit a job to gLite, submits its request to LSF (e.g. using "bsub" command), as he would do for any ENEA-GRID job, but specifies a specific "gLite-interface" queue for the job. In its turn the LSF queue redirects the job towards a special software module that generates a proper jdl file and forwards the job and its jdl to a gLite User Interface. From there it is responsibility of the gLite middleware to send the job to the appropriate Computing Element and report to the interface software module when the job is completed.

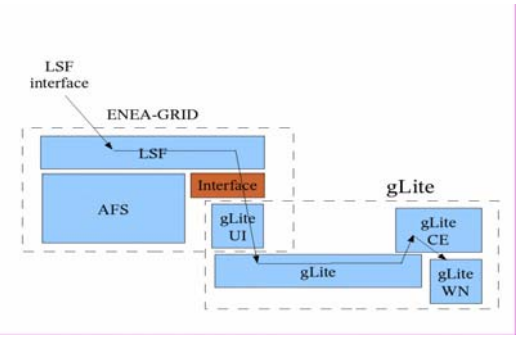


Figure 1: ENEA-GRID to gLite job submission

B. gLite to ENEA-GRID: The SPAGO technique

Allowing job submission from gLite to ENEA-GRID, is, in concept, easier. This is

because the the gLite CE employs LSF as one of its resource managers. Therefore, with proper configuration a CE machine can be interfaced seamlessly with the whole ENEA-GRID infrastructure. However, a different kind of issue arises. The ENEA-GRID infrastructure is very heterogeneous, supporting many different HW/OS platforms. Conversely, the gLite mainstream currently supports only PCs with Scientific Linux operating system. Additionally it requires to install the gLite middleware on all the machines, which is an unrealistic requirement for ENEA-GRID. Therefore a novel technique has been introduced to cope with the issues. The technique is named Shared Proxy Approach for Grid Objects (SPAGO).

The basic design principle of SPAGO is outlined in Figure 2 and it exploits the possibility to use a shared file system. When the CE receives a job from the WMS, the gLite software on the CE employs the resource manager to schedule jobs for the various Worker Nodes, as in the standard gLite architecture.

However the worker node is not capable to run the gLite software that recovers the InputSandbox. To solve this problem the LSF configuration has been modified so that any attempt to execute gLite software on a Worker Node actually executes the command on a specific machine, labeled Proxy Worker Node which is able to run standard gLite.

By redirecting the gLite command to the Proxy WN, the command is executed, and the InputSandbox is downloaded into the working directory of the Proxy WN.

The working directory of each Grid user is maintained into the shared filesystem, and is shared among all the Worker Nodes and the Proxy WN, thus downloading a file into the working directory of the Proxy WN makes it available to all the other Worker Nodes as well. Now the job on the WN1 can run since its InputSandbox has been correctly downloaded into its working directory. When the job generates output files the OutputSandbox is sent back to the WMS storage by using the same method.

In the above architecture, the Proxy WN may become a bottleneck since its task is to perform requests coming from many Worker Nodes. In that case a pool of Proxy WN can be allocated to distribute the load equally among them.

III. Challenges and Solutions

In the last year gLite has made the transition from version 3.0 to 3.1. The main difference between the two versions was the fact that gLite 3.1 supported Scientific Linux 4, instead of the Scientific Linux 3 of the previous version. However, as a side effect some components of the gLite CE underwent a serious structural re-design, with almost no documentation. This change affected the SPAGO components residing on the CE at two different levels:

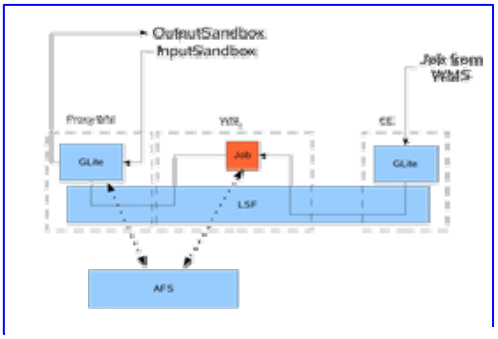


Figure 2:gLite to ENEA-GRID job submission

A. Job filtering.

In gLite the jobs received by the CE are wrapped into a script whose task is to properly setup the environment for job execution. However this script is tailored to work on worker nodes employing SL4. In a SPAGO

environment this script is deployed on a worker node with many possible operating system and we have had issues in which it would not work on other UNIX operating systems. Therefore in the SPAGO approach we modify on the PERL scripts that interfaces with LSF, filters the job and produces the wrapper. In the transition from gLite 3.0 to 3.1, the job filtering mechanism has been deeply modified, thus we had to work on a new version.

B. Authentication.

In gLite the authentication mechanism relies on X509 certificates[4]. This is usually not a problem for usage of SPAGO, but the specific shared file system employed in ENEA-GRID, namely AFS, requires his own authentication tokens to work properly. In gLite 3.0 there was a plugin that allowed seamless conversion between the X509 certificates and the AFS tokens. However the support to this plugin has been discontinued with gLite 3.1 making authentication impossible. Therefore part of our work has been to develop an additional daemon, residing on the CE, that can recover the tokens for the pool accounts as needed.

IV. Interoperability with other projects

A. CRESCO project.

CRESCO (Computational Research Center for Complex Systems) is an ENEA project co-funded by the Italian Ministry for University and Research (MUR). The project is built around an HPC platform, through the creation of a number of scientific thematic laboratories:

1) The Computing Science Laboratory.
2) The Computational Systems Biology Laboratory.
3) The Complex Networks Systems Laboratory.

The HPC platform consists mainly of two Sections: (i) a first section for jobs with limited parallelism, consisting a toal amount of 672 cores; (ii) a second section for high parallelism jobs, consisting in 256 IBM HS21 blades each with 2 Xeon processors for a total of 2048 cores. The machines of each section are connected using an InfiniBand[5] network 4X DDR; also, InfiniBand interconnects the two separate sections, allowing them to operate as a single cluster of 2720 cores. All the cores share a GPFS shared filesystem of hundred of Tbytes, accessed through InfiniBand. Moreover the whole HPC system has been integrated inside ENEA-GRID[1].

B. Interoperability.

In the context of the projects of the PON 1575, there has been an ongoing work attempting to unify the four largest projects (CRESCO, SCOPE, CYBERSAR, PI2S2) into a single interoperable entity. The aim is to realize a single Grid between the four research centers in south Italy, called GRISU. In order to achieve the interoperability the components of the project have chosen gLite as their standard interconnection middleware. However integrating CRESCO machines into gLite is difficult, because they run Red Had 5.1 as their operating system, which is not yet supported by gLite.

In order to solve the problem, CRESCO has been interfaced to the other PON projects by using the SPAGO technology. This also avoided the installation of additional software on the computational cores, which might have interfered with the software already requested by the virtual laboratories. More in detail, gLite services, like the site-bdii, hlr, etc, have been deployed on gateway machines, not directly involved with the HPC system. The core services (CE, his related proxyWNs, and SE) have been deployed on front-end machines, which are not part of the computing cores, but are interconnected to it by InfiniBand and GPFS. The CE and proxyWN machines submit jobs to CRESCO cores using the SPAGO

technology, where GPFS is the shared filesystem (instead of OpenAFS, used in other parts of ENEA-GRID). In the same way, the SE will store the data on GPFS instead than on his local storage.

V. Conclusion

The recent research activities of ENEA, and particularly of CRESCO project, have focused on interoperability. Interoperability with the EGEE project products (gLite), and interoperability with other PON 1575 projects. This has been realized by taking the gateway approach, mainly based on SPAGO technology. The work described in this paper shows that this technology is relatively robust despite software evolution and can be easily implemented by choosing its components into a large set of different technologies. The final result of adopting SPAGO allows exploitation of many different platforms not originally supported by gLite.

References

[1] G. Bracco, S. Podda, S. Migliori, P. D'Angelo, A. Quintiliani, D. Giammattei, M. De Rosa, S. Pierattini, G. Furini, R. Guadagni, F. Simoni, A. Perozziello, A. De Gaetano, S. Pecoraio, A. Santoro, C. Sciò, A. Rocchi, A. Funel, S. Raia, G. Aprea, U. Ferrara, D. Novi, and G. Guarnieri, "CRESCO HPC System Integrated into ENEA-GRID Environment", Workshop Finale dei Progetti GRID del PON "Ricerca" 2000-2006 – Avviso 1575, 2009.

[2] Bracco, G; Migliori S.; Quintiliani A. ; S. Sciò, C. ; Santoro , A**, "**The ENEA gateway approach to provide EGEE/gLite access to non-standard architectures**",** EGEE'07 Conference, 2007.

[3] EGEE, "gLite", http://glite.web.cern.ch/glite/

[4] IETF, "Internet X.509 public key infrastructure", RFC3280, 2002.

[5] InfiniBand Trade Association, "InfiniBand Architecture overview", http://www.infinibandta.org/events/past/it_roadshow/overview.pdf

[6] MIT Kerberos Consortium, "kerberos: the network authentication protocol", http://web.mit.edu/Kerberos/

[7] "OpenAFS", http://www.openafs.org/.

[8] Platform Computing, "Platform LSF Multicluster", http://www.platform.com/Products/platform-lsf-multicluster

[9] A. Santoro, G. Bracco, S. Migliori, C. Scio, A. Quintiliani, A. Rocchi, "A simple SPAGO implementation enabling the connection of heterogeneous computational resources to a gLite based Grid", EGEE'08 User Forum, 2008.

[10] A. Santoro, G. Bracco , S. Migliori, S. Podda, A. Quintiliani, A. Rocchi, C. Sciò, "Integration of heterogeneous computational resources in EGEE infrastructure: a live demo", EGEE'08 Conference, 2008.

Andrea Santoro received a ``Laurea" degree (MS level) in Electronic Engineering from the University of Rome "La Sapienza", and a Ph.D. in Computer Engineering from the same institution. He worked at Georgia Tech in 2004, as a post-doc lecturer in the University of Rome "La Sapienza", (entrusted with full responsibility on two graduate-level courses) until 2007, and is now a research assistant at the Italian agency for new technology, energy and environment (ENEA). He is an IEEE Member and his research interests are in the fields of parallel simulation, operating systems, modeling and networks. He published articles on those fields on IEEE Transactions, other journals and several international conferences. Additionally he serves as a referee for international conferences and important journals in the same areas. His e-mail address is: santoro@dis.uniroma1.it and his web page is http://www.dis.uniroma1.it/~santoroa.

The gLite Workload Management System

M. Cecchi[1], A. Dorigo[1], A. Ghiselli[1], F. Giacomini[1], A. Maraschini[2], M. Marzolla[2], S. Monforte[1], L. Petronzio[2] and F. Pacini[2]
I.N.F.N. - National Institute for Nuclear Physics - Viale Berti Pichat, 6/2 - Bologna Italy
Elsag-Datamat s.p.a. - Via Laurentina, 760 - Roma, Italy

***Abstract*—The gLite Workload Management System (WMS) represents a key entry point to high-end services available on a Grid to provide reliable and efficient distribution and management of end-user requests. This service basically translates user requirements and preferences into specific operations and decisions dictated by the general status of all other Grid services, while taking responsibility to bring requests to successful completion. This is accomplished interoperating with a number of other Grid services, especially with Logging and Information System. Thanks to the modularity of its design, the WMS can be deployed in different layouts and scenarios according to specific needs of Grid or Virtual Organization administrators. Several added value features are provided for job submission, different job types are supported from simple batch to a variety of compounds. Exploiting specific attributes for defining groups and roles, assigned by the VO managers to their users, the WMS is able to deal with authorization policies for computing and data access in a heterogeneous resource environment. In this paper, we outline what has been achieved to provide adequate workload and management components, suitable to be deployed in a production-quality Grid, with particular mention to newly introduced features and reached performance.**

I.INTRODUCTION

GRID computing technologies have been developed over the last decade to provide a computing infrastructure for a disparate and ever growing number of e-Science applications. A first large scale production Grid infrastructure was deployed by the Enabling Grids for E-SciencE (EGEE) [1] EU-funded project. Its operation was then further consolidated during its second phase (EGEE-II). The EGEE Grid infrastructure consists of a set of middleware services deployed on a worldwide collection of computational resources, with an extensive programme of middleware re-engineering that has resulted in a consolidated software stack, gLite [2]. This long-standing project, now at its third phase (EGEE-III), will take further steps in moving Grids to dependable and sustainable production infrastructure while providing a continuous service to its expanding user base. EGEE-III will continue to develop gLite as its reference open-source middleware distribution.

In this paper we outline what has been achieved to provide adequate workload and management components, suitable to be deployed in a production-quality Grid, while covering the design and development of the gLite WMS, with particular respect to functionality and interoperability, focusing on the most recently achieved results.

II. FUNCTIONALITY

The gLite Workload Management System (WMS) provides a service responsible for the distribution and management of tasks across resources available on a Grid, in such a way that applications are conveniently, efficiently and effectively executed. These tasks, which

basically consists in execution requests, are usually referred to as "jobs". In a Grid environment the scope of such tasks/jobs needs to be extended to take into account other kinds of resources, such as storage or network capacity. The need for such a broader definition is basically due to the move from typical batch-like activity to applications with ever more demanding requirements in areas like data access or interactivity, both with the user and with other tasks.
In this respect, the WMS does support different types of jobs:

- Single batch jobs
- Work-flows: jobs with dependencies expressed as a direct acyclic graph (DAG)
- Collections: sets of jobs without dependencies grouped together and identified by a single handler
- MPI: based on message passing interface - a widely-used library to allow for parallel programming within a single cluster (intra-cluster)
- Interactive: establishing a synchronous two way communication with the user on a socket stream
- Parametric: allowing multiple jobs to be defined by a single description with attributes varying with a parameter

The characteristics of a job are defined using a flexible and expressive formalism called Job Description Language (JDL) [3]. The JDL is based on Classified Advertisements or *ClassAds* [4], developed within the Condor project [5], which basically consist of a list of key/value pairs that represent the various characteristics of a job (input files, arguments, executable, etc.) as well as its requirements, constraints and preferences (physical and virtual memory, CPU, operating system, etc.). The user can then specify whatever attribute for the description of a request without incurring in formal errors, as ClassAds are not bound by any particular schema. Only a certain set of attributes are directly taken into account by the WMS on the base of documented semantics, the others will simply be passed on without specific processing. Also, the attributes used for describing high-end resources come from a common schema, the so called GLUE schema [6], born from a joint effort to standardize and facilitate interoperation between Grid infrastructures. Jobs are always associated with user proxy credentials and all job-dependent operations are performed on behalf of the user. gLite in general and the WMS in particular exploit experience and existing components from the Virtual Data Toolkit from Condor and Globus [7]. While Condor plays a significant role in the present architecture as a job submission and tracking layer (see later), the Globus Security Infrastructure (GSI) is used throughout for enabling secure authentication and communication. GSI provides in fact libraries and tools for authentication and message protection that use standard X.509 public key certificates, public key infrastructure (PKI), the SSL/TLS protocol and X.509 Proxy Certificates, an extension defined for GSI to meet the dynamic delegation requirements of Grid communities. A specific service, called Proxy Renewal and conceived as be part of the WMS, is devoted to renewing credentials, automatically and securely, for long-running jobs.

The Grid is a complex system and things can go wrong at various stages of the so called submission chain. The WMS has been designed with the ability to recover from failures of the infrastructure by automatically resubmitting failed jobs, this is done at two levels. "Shallow" resubmission is utilized in those cases where an error occurs before the CE has started executing the job, in which case another CE can be tried immediately without any worry to compromise the results. This will also reduce the probability to have multiple instances of the same job over the Grid due to temporary loss of network contact. "Deep"

resubmission happens whenever a job fails after it started running; this situation can be more problematic as the job may well have done a considerable amount of processing, producing output files or making other state changes, and may also have consumed a significant amount of (precious) CPU time. Users can therefore choose the number of times they will allow the job to be resubmitted in these two ways with two parameters of the JDL. If a job fails after having reached the maximum number of retries it will be terminally aborted. Submitting a job actually means passing its responsibility to the WMS; its purpose is finding the appropriate resource(s) matching the user requirements, watching and directing the job on its way to completion, with particular attention to infrastructure failures requiring resubmission. The WMS will in the end forward the job to the selected set of CEs for execution. The decision about which resource is adequate to run the job is the outcome of a so called match-making process between the "demand", represented by the submission requirements and preferences, and the "offer", represented by the characteristics of the available resources. The availability of resources for a particular task depends not only on the actual state of the resources, but also on the utilization policies that the resource administrators and/or the administrator of the Virtual Organization (VO) the user belongs to have defined for each of their users.

It can happen, not rarely, that none of the resources available on the Grid at a given is able to satisfy some job's requirements, in such case the submission request is kept pending by the WMS and periodically retried, the retry period being a configuration parameter, until the request expires.

Besides request submission, the WMS also implements request management and control functionality such as cancellation and output retrieval. Another feature exists to list all the available resources matching a given job at a given time so that if a user (which can, by the way, also be represented by an automatic system) has no matching resources it can temporarily stop submitting. Request status follow-up can be achieved through the Logging&Bookeeping service (L&B) [8], another key service responsible for tracking jobs in terms of events (important points of job life, e.g. submission, transfer from a WMS component to another one, finding a matching CE, starting execution etc.) gathered from various WMS components as well as other Grid services. Each event type carries its specific attributes. The entire architecture is specialized for this purpose and is job-centric: any event is assigned to a unique Grid job identifier. The events are gathered from various WMS components by the L&B producer library, and passed on to the local logger daemon, running physically close to avoid any sort of network problems in a store&forward fashion.

All the various job management tasks mentioned so far are accomplished by different components basically implemented (mostly in C++, with extensive usage of the Boost [9] libraries) as different processes or threads, all communicating via persistent data structures. As anticipated, one core component is the Match-Maker which sorts out a list of resources satisfying the given requirements.

These resources might even include Storage Elements (SE, i.e. the Grid abstraction for a storage resource) if the user requested to need manipulate data. Such returned list of suitable resources is ordered, given that more than one resource could match the specified requirements. To avoid the top-ranked resource to be repeatedly chosen upon successively close in time requests, so becoming overrated, before the Local Batch System and the Information Systems could in turn update such dynamic information, a stochastic algorithm can be used to perform a smoothed selection among all the matching resources - weighted according to their actual rank - in such a way to

prevent congestion for the initially best ranked resources. Proper handling of massive data volumes is a very important aspect in production quality Grids (it is maybe worth noting that one of the projects from which EGEE originates was called "DataGrid"). The JDL allows the definition of requirements based on data through an attribute called "DataRequirements" which is structured in such a way to allows users to target experiment-specific catalogs for their jobs and to mix different input data types supported by different data catalogs in the same job description. Logical File Names (LFN), Grid Unique ID-entifiers (GUID), Logical Dataset (LDS) and/or generic queries can be used to retrieve data from SEs. All of them are used by the WMS to query the related Data Catalog for getting back a list of Physical File names (PFN) that are needed by the job as input for processing ([10] for more information). Output data can then be stored to a specified SE and registered to a catalog. While match-making is made between two entities - typically the job and the computing resource, another interesting feature relating data management, called gang-matching allows to take into account, besides CE information, also SEs in the process. Another interesting feature, which has been added quite recently, is represented by the so called "bulk match-making". This optimization, enabled for collections, allows match-making for each subset of jobs sharing the same characteristics instead of matching each single job. The original collection is partitioned into such subsets according to some significant attributes (JDL attribute "SignificantAttributes") which will identify by the equivalence classes. A typical use-case for specifying significant attributes could be, as an example, parting the original set on "Requirements", "DataRequirements" and "Rank". Here is a summary of the more relevant functionalities implemented in the gLite WMS:

- resubmission: shallow or deep;
- stochastic ranking;
- bulk-submission and bulk match-making;
- proxy renewal;
- support for MPI jobs even if the file system is not shared between CE and Worker Nodes (WN);
- support for execution of all DAG nodes within a single CE - chosen by either user or by the WMS match-maker;
- support for file peeking to access files during job execution;
- load limiting mechanism to prevent system congestion based on machine's vital parameters;
- automatic sandbox files archiving/compression and sharing between jobs;
- match-making with data;
- gang-matching.

III. INTEROPERABILITY

Given the typically large number of different parties involved in a Grid infrastructure, interoperability plays a key role to facilitate establishing and coordinating agreements and interactions between all the involved entities. In this respect, the WMS, especially by virtue of his central, mediating role, has to deal with a wide variety of people, services, protocols and more, ranging from users - belonging to different VOs - to other services of the EGEE/gLite infrastructure and to other Grids as well.

For what concerns users, to be able to allow interaction adhering to the SOA model, a Simple Object Access Protocol (SOAP) Web Service has been implemented, its interface being described through a Web Service Description Language (WSDL) specification written in accordance to the WS-I profile, which defines a set of Web Services specifications to promote interoperability. This newly introduced Web Service based

implementation replaced a legacy network interface based on a proprietary protocol.It manages user authentication/authorization and operation requests. It runs in an Apache [11] container extended with FastCGI [12] and Grid Site [13] modules. The Fast CGI module implements Common Gateway Interface (CGI) functionality along with some other specific features. The most important advantages of using FastCGI are its performance and persistence. FastCGI applications, in fact, are able to serve, in a multiprocessing fashion, multiple requests, where instances can be dynamically spawned or terminated according to the demand. In particular, an additional control mechanism over unpredictable error conditions such as undefined hanging has been implemented to automatically terminate a serving process of the pool after a given configurable number of requests. Moreover, the Grid Site module provides an extension to the Apache Web Server for use within Grid frameworks by adding support for Grid Security Infrastructure (GSI), the Virtual Organization Membership Service (VOMS) [14] and file transfer over secure HTTP. It also provides a library for handling Grid Access Control Lists (GACL). The Web Service hosting framework provided by Apache, Grid Site and gSOAP has allowed the development of this front-end interoperable service in C++, giving continuity and consistency with the rest of the coding. About interoperation with other Grid services, we need to describe in more detail how job management is accomplished by the WMS.

A service called Job Submission Service (JSS) is responsible to actually establish an authenticated communication with the selected resource to forward the job and to monitor its execution. To implement such lower level layer Condor-G has been always adopted. A monitoring service, part of the JSS, is also responsible for watching the Condor log files intercepting interesting events concerning active jobs which affect the job state machine and trigger appropriate actions. Every CE supported by Condor-G is then implicitly supported by the WMS as well, in particular the LCG CE (pre-Web-Service Condor-G plus GRAM on the CE) and the gLite CE (pre-WS Condor-G plus Condor-C on the CE). Recently, with the advent of the newest WS-I/BES [15] CE called CREAM [16], a new component of the WMS suite, called Inteface to CREAM Environment (ICE), has been introduced as part of JSS for job management towards CREAM. ICE is a gSOAP/C++ layer which will securely manage job operations to CREAM CEs. In doing so, it subscribes to the gLite CEMon information system [17] in order to asynchronously receive notifications about job status changes. ICE also performs synchronous status polling for unresponsive jobs, in case some notifications are lost. Interoperation with Information Providers is achieved either synchronously or asynchronously for those providers who support it. We actually do provide interfacing with the Berkely Database Information Index (BDII), support for other providers has been recently dismissed due to lack of use.

About formalisms for defining jobs, the WMS fully endorses the Job Submission Description Language (JSDL). This is an emerging OGF standard which aims at facilitating interoperability in heterogeneous environments, through the use of an XML based job description language that is free of platform and language bindings. JSDL contains a vocabulary and normative XML Schema that facilitate the expression of job requirements and preferences as a set of XML items. What happened in the past and still can happen is that several different organizations accommodate a variety of job management systems, where each system has its own language for describing job submission. This represents a severe obstacle for interoperability. In order to utilize such different systems altogether the involved organizations would have to prepare and maintain a number of different job submission documents, one for each system, basically all describing the same operations. The JSDL represent a significant effort toward unification

and has semantics comparable to the current ClassAd-based JDL, its adoption as an OGF approved standard makes it a good candidate for support by the WMS.

On the front of Grid interoperability, having already set up a long-standing interaction with OSG, recent work has been done to enable interoperability with both NorduGrid [18], and its ARC CE, and UNICORE [19], with a contribution to the writing of the Grid Interoperation Now (GIN) [20] profile.

More pragmatically, much of the issues concerning interoperability reflects in the way the WMS job-wrapper (the shell script generated by the WMS which surrounds the user job execution and performs basic setup and cleanup operations, downloading/uploading the sandbox, setting the execution environment, logging etc.) is engineered. Due to the diverse nature of resources belonging to one or more Grids, such script must be kept as simple and as robust as possible. The job-wrapper script may be running in an unfriendly WN environment where no or little assumption can be made on what is available. Again, due to the pivotal role of this script, a significant work has also been done to extend it in order to encompass all the different requirements expressed by the involved parties (users, VOs and resources) without losing functionality nor generality. To achieve this, a series of hooks is provided in the job wrapper generation procedure, allowing specific customisations to be inserted by users, VO managers and site administrators. This approach reduces hard-coding, by decoupling general and specific operations, without limiting functionality. For users, prologue and epilogue scripts have been included - to be run before and after the job is executed - basically with the intent of setting and cleaning up the proper environment for "real" jobs; for VOs, a customisation point is foreseen mostly used to hook up the proper middleware version; for similar purposes resource managers are allowed to hook up their scripts throughout several strategic points of the job-wrapper.

Here is a summarized list of the functionality provided in the areas of integration with other services and interoperability:

- backwards compatibility with LCG-2;
- automatic renewal of credentials;
- gridFTP and HTTPS to handle secure file transfer for the sandbox;
- service discovery for obtaining new service endpoints to be contacted;
- support of different mechanisms to populate the ISM from several sources (BDII, R-GMA, CeMon);
- support for submission and monitoring for the LCG, gLite and CREAM CEs;
- support for Data management interfaces (DLI and StorageIndex);
- support for JSDL;
- support for Grid Site delegation 2.0;
- interoperability with the American Open Science Grid (OSG), Nordugrid and UNICORE;
- integration with Grid accounting and authorization frameworks;
- user prologue/epilogue scripts accompanying the job, more custom scripts allowed to be hooked for use by resource and VO administrators.

IV. RESULTS AND FUTURE DEVELOPMENTS

As of late 2008, the WMS has been deployed in a large number of multi-user and multi-VO scenarios. Thorough testing and intense troubleshooting have been accomplished during all these years, of course driven by the compelling needs of the LHC experiments. This has led to a significant level of service stability for the current production release. Much of this effort was accomplished using the development test-bed and the preview test-bed,

which also includes new components not yet ready to be deployed in production (as it was the case for ICE). In addition, the concept of Experimental Service proved to be very effective: a development instance, attached to the production infrastructure, to be accessed by a selected number of users and immediately installed with the latest available patches. Now that an acceptable level of sustained stability has been reached, work is being done to further target performance. In particular, after the (effective) introduction of collections, the average match-making time, performed on the full production BDII, has room to improve, especially for single jobs. The next to come release will be able to perform the match-making in parallel thanks to a re-design of the ISM that will be doubled in order to remove some locks with a huge scope at the moment necessary to keep the structure synchronised with readers and writers insisting on it. A read-only copy will be available for readers (the request handlers needing to perform the match-making) while another one will be created in background while purchasing. A pseudo-atomic swap between these two copies will occur periodically and timely so that the ISM at the moment accessed by reader threads is disabled while, in the mean-time, the freshly purchased one (since then only accessed for writing) will then become available to readers (only). Two ISM instances will be contemporarily present in memory only for limited period - the time needed to carry out purchasing and to wait for the older threads, still pointing to that very copy, to complete - after which such instance can be definitely cleared. Such a design has already been stress tested in a prototypal instance installed as an experimental service, with the collaboration of the CMS experiment; a peak performance of about 100.000 jobs/day were reached for more than two consecutive days. The next gLite WMS release (3.2), under preparation at the time of writing, will contain several improvements, as the result of the intense restructuring activity which took place during EGEE-II, not yet fully ported to the release branches. Among other things, this new release, aimed at providing a more lightweight and responsive service thanks to a significant redesign of its core component, will be instrumented with the aforementioned parallel match-making, IPv6 compliancy, support for Grid Site delegation 2.0 and it will be Scientific Linux 5 ready. An official Web Site and Twiki pages have been set up, being kept readily updated, for documentation and support about all the activity concerning the WMS.

V. CONCLUSION

The gLite WMS is designed and implemented to provide a dependable, robust and reliable service for efficient distribution and management of end-user requests for computation, storage, network, instruments and whatever resource may be shared across a production quality Grid. It comes with a fully-fledged set of added-value features to enhance low-level job submission. Thanks to the flexibility of a scalable, fault-tolerant and service-oriented architecture it has been deployed in a number of layouts and scenarios. After several years of operation the WMS has reached sustained stability and a performance targeted at covering the current needs, coming in particular way from High Energy Physics and Bioinformatics. Development continues by supporting enhancements requests expressed by the increasing number of its users and keeping up with compliancy to emerging formal and de-facto standards and protocols. We will also continue facing the challenge of reaching even higher levels of performance, scalability and reliability to find us prepared to meet the growing demand of the EGEE infrastructure.

REFERENCES

[1] http://www.eu-egee.org/
[2] http://glite.web.cern.ch/glite/
[3] JDL Attributes Specification, https://edms.cern.ch/document/590869/1,

EGEE-JRA1-TEC-590869-JDL-Attributes-v0-4
[4] http://www.cs.wisc.edu/condor/classad/
[5] M.J Litzkow, M. Livny and M.W. Mutka, *Condor-A hunter of idle workstations*, Proceedings of the 8th International Conf. On Distributed Computing, San Jose, CA USA (1988) 104-111
[6] www.globus.org
[7] http://forge.gridforum.org/sf/projects/glue-wg
[8] F. Dvorak, D. Kouril, A. Krenek, L. Matyska, M. Mulac, J. Pospisil, M. Ruda. Z. Salvet, J. Sitera, J. Skrabal, M. Vocu et. al., *Services for Tracking and Archival of Grid Job Information*, CGW05, Cracow - Poland, November 20 - 23, 2005
[9] www.boost.org
[10] https://edms.cern.ch/document/487871
[11] http://www.apache.org
[12] http://www.fastcgi.com
[13] http://www.gridsite.org
[14] V. Chiaschini et al., An Integrated Framework for VO-oriented Authorization, Policy-based Management and Accounting}, Computing in High Energy and Nuclear Physics (CHEP'06)), T.I.F.R. Mumbai, India, February 13-17, 2006
[15] http://grid.pd.infn.it/NA5/bes-wg.html
[16] P. Andreetto, S. A. Borgia, A. Dorigo, A. Gianelle, M. Marzolla, M. Mordacchini, M. Sgaravatto, L. Zangrando et. al., *CREAM: a simple, Grid-accessible, job management system for local computational resources*, Computing in High Energy and Nuclear Physics (CHEP'06)), T.I.F.R. Mumbai, India, February 13-17, 2006.
[17] http://grid.pd.infn.it/cemon/field.php
[18] http://www.nordugrid.org
[19] http://www.unicore.eu/
[20] http://forge.ogf.org/sf/projects/gin

New ENEA-GRID Integrated Support Services for CRESCO

Alessio Rocchi[1], Giovanni Bracco[1], Silvio Migliori[1], Salvatore Podda[1], Andrea Quintiliani[1], Andrea Santoro[1], Carlo Sciò[2]
[1]*ENEA - Italian National Agency for New Technologies, Energy and Environment - V. E. Fermi 45, 00044 Frascati (Rome), Italy, alessio.rocchi@enea.it*
[2]*Esse3Esse, Rome, Italy*

***Abstract*—The ENEA-GRID environment is based on three fundamental pillars, the Andrew File System (OpenAFS, providing facilities for the resource distribution over a WAN), the Load Sharing Facility (LSF, capable of handling the resource and load management) and a Kerberos 5 authentication realm. CRESCO HPC center is the main resource at the moment. In the CRESCO framework, many support services have been designed to improve both the user experience and the administrative maintenance tasks. On the administrative front, we have engineered two applications. (a) AMACA (AFS Memorize and Check Application) is a two-module application aiming to trace the status of every AFS core component, while providing an open, state-of-the-art monitoring platform for both administrators and users. (b) ARCO (AFS Remote Command Operator) is a Service Oriented web-based application, whose purpose is to submit remote commands to machines registered by system administrators. On the user side, the following utilities have been developed: (a) JobRama, aiming to monitor the status of jobs, queues and hosts exposed by LSF. The utility uses the LSF data structures and routines to get informations and deploy them in a simple, fast and adaptive GUI. (b) A ticket system (Xoops based) allowing researchers to be quickly supported in their work, while lightening backend tasks for administrators (c) A web interface for new account requests. (d) A web interface for password change without using command line All the utilities are fully integrated in ENEA-GRID environment and manage the access through a single set of credentials, by means of a Unified Access API (UAA) for the authentication over an OpenAFS cell.**

***Index Terms*—ENEA, CRESCO, HPC, Grid, Grid software services, Grid support tools.**

I. Introduction

THE ENEA-GRID environment is massively distributed over a large geographic area. Information management and distribution in such an extended Grid framework it is complex to handle, both on users and administrators side. Moreover, needs for software applications able to automate tasks or make easier the resources fruition are often unattended: prepackaged tools, when they exist, are generally difficult to be modeled and adapted with low effort. The present document illustrates about some software components we designed and implemented: perfectly tailored on whole Grid system, fully integrated with it, effective, usable, and (possibly) easily deployable to other infrastructures. The section starts describing a technology, fundamental for our integration purposes, which represent the common low-level layer for every other application.

A. Unified Access API (UAA)

The Unified Access API realizes the way whereby a Grid user can access to a web-deployed service using always a single set of credentials (i.e. the username and password stored in Kerberos 5 database). UAA is written in PHP (but it could be easily ported toward

other programming languages) and provides an interface both with Kerberos v5 [1] (providing mechanisms for authentication over the realm ENEA.IT) and OpenAFS [2] (managing the authorization aspects, on the basis of a granular set of Access Control Lists). In particular, OpenAFS provides an advanced authorization schema, based on so-called PTS groups. Architecture is very simple and has two phases: first, user credentials are sent to Kerberos for validation. If authentication is successful, a Kerberos ticket is produced, and UAA checks for a valid PTS membership. Such control can be performed both directly (for example when user tries to access to a shared folder), or on the basis of a pre-compiled list of allowed groups (usually when UAA is on the frontend of a web application). If the second step outcome is positive, OpenAFS stores in the kernel memory a set of data called "token". At any time, only one token per cell can exist (so that subsequent authentication requests cause latest token to overwrite the previous one). It is essential to guarantee that each user session will have its own token. Concurrency management and race conditions avoidance over token acquisition is granted by implementing Inter Process Communication with Process Authentication Group (PAG) shell (an OpenAFS specific feature), that provides the essential isolation layer. We have build two distinct implementations of access mechanism lying on top of UAA: the first is thought to protect web applications, as it provides standard ways to memorize the state of HTTP transaction by means of cookies, while the other one is made available in order to safeguard "classic" web pages (or, generically, static resources exposed through a web server). Just this latter is built on top of Jan Wolter's mod-auth-external module [3], conveniently patched, and so it is fully embedded with Apache [4]. Relying on UAA brings many advantages: administrators and developers do not need to deploy many accounting database (one per application, in the worst case), stepping up security and maintainability; UAA grants granularity on access discrimination, because authentication procedures can also validate group membership. One immediate result is the faculty to give applications root or admin rights when needed. The UAA approach has some limitations: as it relies on OpenAFS's Unix specific features, it can work only on these platforms. It needs also to be carefully managed by the system administrator because PHP IPC features have to be unlocked.

II. Administration services

A. AMACA (AFS Memorize And Check Application)

The administration of OpenAFS is more complex than the one requested to manage ordinary POSIX filesystems. When dealing with information distribution over a Wide Area Network, OpenAFS transfers data and metadata whereof is important to keep track. For example, is helpful to know when a certain modification has been applied or some change has occurred. Such a need requires to implement a historical database, that must be also able to provide data on file system consistency. Utilities performing this kind of checks could be useful (especially when dealing with large information sparseness), but are not deployed with the OpenAFS vanilla package. Hence, administrators have to rely only on command line tools to accomplish the maintenance tasks. AMACA (AFS Memorize And Check Application) is a software engineered in response to these requirements. It is an application able to check the status of every OpenAFS core component (programmatically or on-demand) and to show the exploration results on a convenient web interface. Next fig.1 shows the global architecture.

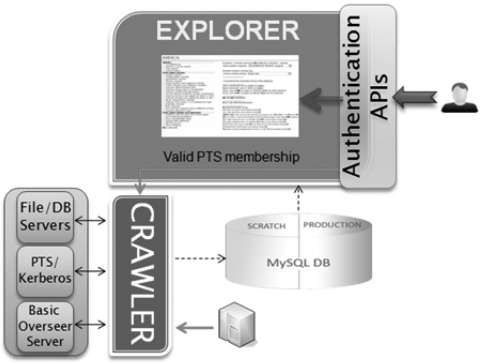


Fig. 1. AMACA architecture

As the previous image shows, AMACA features are carried out by two main modules: the crawler is responsible to perform the full-blown indexing, as the explorer offers the way to easily get back the results and analyze them. Crawler module is written using the Norbert Gruener's OpenAFS Perl API [5], that provide a great interface with the Andrew file System internals. Metadata are stored, unstructured, into a MySQL backend. The presence of a database permits is great in order to perform fine-grained data-warehousing operations on gathered data. Differentiation among subsequent crawler invocation is done by identifying every result set with an unique identifier called snapshot. Explorer implements business logic about data aggregation and visualization. Users can navigate through datasets afferent to different snapshots, decide to restrict the view only on a particular domain (rather than to the entire cell), perform date based comparisons among different file system snapshots, execute parameterized queries, accomplish tasks of historical analysis, and so on. One of the most useful features of Explorer is to show alarms about potentially dangerous situations. Given some research patterns (coded in the crawler), if AMACA detects any mapping harmful situation, explorer shows them in the first page. Therefore, administrators will work on solving them. They typically do not have to wait for the scheduler to launch another instance of the crawling module (i.e. the next day) to see the impact of its operations: if a user belong to a valid group (customizable), he can decide to perform an independent crawling whose results will fill only a "scratch", dedicated portion of the database (so that the other part, called "production", will only contain "official" data). AMACA is designed to be extensible, so everyone knowing Perl and Php programming languages can write its own probes and embed them in the crawler code.

B. ARCO (AFS Remote Command Operator)

ARCO is a software able to execute general-purpose commands on any cluster. The way ARCO operates is very simple: administrators define target machines and services to execute on them (single commands or script residing in a shared path, for example under a subtree of the root directory of OpenAFS). Subsequently, an association is created between those services and machines and, from this moment on, it is possible to start executing commands. Due to potentially critical nature of commands, ARCO uses an extended implementation of the UAA, where membership is validated so that ability of users to perform operations is done on the basis of different security levels (there is no limit on hierarchy of groups, as every command requires a membership to be specified in order to be correctly registered). Commands are sent to target machines by means of features provided by *libssh* [6] (as SSH connection is done using Diffie-Hellman's key exchange method, there is the need to be sure that the key file is stored in a secure place). A MySQL backend is used to keep track of both commands output (both stdout and stderr are traced), and to log activities performed by any user until logout occurs. Security is guaranteed by two cryptography layers: the passphrase (needed during the key exchange) is stored encrypted in the database (using algorithms provided by *mcrypt* library). Moreover, to avoid data interception while a command passes the web through HTTP, transactions are performed only through SSL

protocol.

C. WARC

WARC (The web portal of the ARC tool) is a powerful application, developed by Ruggero Nepi from CASPUR in strict collaboration with ENEA, whose purpose is to provide web management of users and project areas defined over a WAN distributed OpenAFS cell. The main advantage in using WARC is that it can delegate administration privileges to selected users. Such privileges can be restricted to one single site, and themselves can be conveniently limited when dealing with project areas. WARC uses a Perl implementation of UAA for authentication purposes.

III. USER SERVICES

A. JobRama

JobRama is a web application whose main purpose is to monitor status of jobs submitted by users to the ENEA-GRID HPC clusters. JobRama is born to operate in a scenario where the only way to get information about a job is to use the shell commands shipped with LSF. It all is generally uncomfortable for users, especially when they are not used to unix systems (which are the native environment for LSF command line tools), or need smarter details about the task they have ran. Moreover, shell commands do not provide all available data. JobRama oversteps those limitations, providing a comfortable GUI where plain and aggregate jobs data are displayed. The main advantage of JobRama is to operate at the same level of LSF shell commands (Fig. 2)

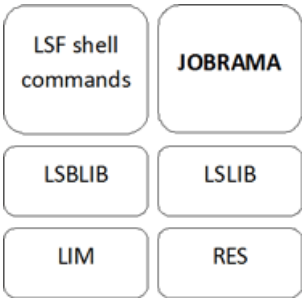


Fig. 2. Jobrama and LSF Stack

The previous image represents where JobRama is placed with respect to the LSF programming stack: at the very lower level we find the job manager daemons (which are responsible for managing resources and load, host selection, job dispatching, etc.). In the middle there is the base and batch API. The latter is used by command line tools in order to submit, control, manipulate and queue jobs[7]. It is clear that JobRama does not display data simply acting as a wrapper for the LSF shell commands. On the contrary, it operates at their same level and uses their same API, with the benefit that it can take advantage from the whole dataset that daemons provide at every instant. The JobRama architecture is represented in the following fig. 3.

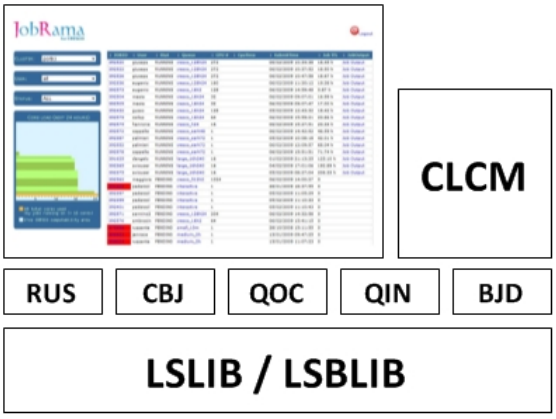


Fig. 3. Jobrama architecture

As previously said, JobRama is built on top of LSF API. The image above makes also clear,

anyway, that changing the lower level API with the ones provided by other queue managers will result in a software still perfectly working; for example it could be easily extended to support Maui Personal Batch System. In Jobrama, the web application at the top level communicates with some external modules whose purpose is to execute native code, on page loading, in order to gather the data which will form the web layout. Specifically, modules at intermediate level manage analysis on resource occupation (RUS), job details (CBJ and BJD), and queues information (QIN and QOC). Every information in this set can be plain (e.g.: job identifier, start date, execution hosts, etc.) or aggregated (e.g.: job time to live, alarms on job in critical state, etc.). Another form of data aggregation is performed by a module (CLCM) working at the same level of the web application (but not belonging to the layout classes): it gathers details on job submission time and combines them with queue policies informations, in order to get a chart of the system load normalized to twenty-four hours. In order to improve experience and usability, web application shows everything on a unique clear page, and allows users to restrict view only to data they are involved with.

B. Ticketing system

Helpdesk for Grid users is managed by a customized implementation of the well-known Xoops Content Management System. Customization has been necessary to improve user registration procedures, and to include UAA support for integrated authentication.

C. Account request manager

This application implements a centralized account request form for ENEA-GRID. The main advantage is in its integration with the helpdesk system: every registration request results in a ticket submission.

D. Password changer

This application allows user to change their password in a visual way. It is essentially a web wrapper of *kpasswd* OpenAFS built-in command, using libexpect[8] to fulfill the data requests. Moreover, a mail mechanism has been implemented (this can't be done by vanilla *kpasswd*) in order to inform users that their password has been changed.

IV. CONCLUSION

The software tools presented in this paper improve the usability and manageability of the ENEA-GRID services. Their adoption has permitted to facilitate the administration, solving the issues connected with the WAN characteristics of the infrastructure. Also non administrative users can benefit of them, as they can get advantage of visual tools to carry on their work. Every software appearing in this document is available for download and freely distributed.

REFERENCES

[1] *MIT Kerberos*, http://web.mit.edu/Kerberos.
[2] *OpenAFS*, http://www.openafs.org.
[3] Jan Wolter, *Apache Mod Auth External*, http://unixpapa.com/mod_auth_external.html.
[4] Norbert Gruener, *OpenAFS Perl API*, http://search.cpan.org/ nog/AFS-2.6.1/.
[5] *LibSSH*, http://0xbadc0de.be/wiki/libssh:libssh.
[6] *Platform LSF Architecture*, http://www.vital-it.ch/LSF/programmer/introduction.html#96101.
[7] *PECL Expect package* - http://pecl.php.net/package/expect.

Alessio Rocchi holds a degree in Computer Engineering from University of Rome "Tor Vergata". After almost 10 years spent as freelance consultant, since 2007 he works as software engineer in ENEA, focusing on support tools.

Interdependencies Among Critical Infrastructures

Silvia Ruzzante[1], Giordano Vicoli[2]
[1]*ENEA-FIM CAMO, Portici Research Center (Naples), Italy, silvia.ruzzante@portici.enea.it*
[2]*ENEA-FIM CAMO, Casaccia Research Center S. Maria di Galeria (Rome), Italy, giordano.vicoli@enea.it*

Abstract — The main goal of the Sub Project CRESCO SPIII.5 has been the study of the phenomena of all the interdependencies among infrastructures with the approach of *What if*, by creating the integration of different simulation codes that represent the functioning of every infrastructure, and by developing a Software Platform (located at ENEA Portici Research Center) in order to analyze the interdependencies among different infrastructures.
The situation is the following: final users enter the services provided by the CRESCO Software Platform by Internet, and after creating an own account, they should set their own simulation requests and send all of them to the Software Platform.
The Software Platform is based on a distributed architecture that allows the communication among different tools that runs into the ENEA GRID and is called **ASSOCIA** (**A**pplication **S**erver for **S**imulation **O**f **C**ritical **I**nfrastructures and data **A**daption). It consists of two kind of middleware: the first one is a *request middleware* that manages all users accounts and requests, and the other one is a middleware that manages the interactions between domain simulators and simulation models (*management middleware*).

Index Terms **— Interdependencies, ENEA GRID, ASSOCIA, Ciab, Cisia.**

I. INTRODUCTION

The research into the phenomena of the interdependencies among critical infrastructures has been the main goal of the subproject CRESCO SPIII.5. This topic has been dealt with the approach of "*What if*", giving a result visualization as close to the reality, using also simulation technologies of virtual reality. For this purpose we have an higher level of difficulty for modeling a set of infrastructures rather than a single model network, because we intend to build all possible interactions among different infrastructures. This activity had to answer the following questions:

a) How is made up of the model for a set of different interdependent complex networks.
b) What is the burden of spatial and economical parameters of the interdependencies, that is how is it possible to design stand-alone networks.
c) What is the contribute of the cyber layer for the networks interdependencies and how to evaluate joined risk.

Dealing with the problem

The research goal is to create the integration of different simulation codes studying interdependencies among infrastructures and giving information about it to accountable people of critical situations. So, we can note that a probable Simulation Center takes care of only the analysis of the interdependencies

among all the infrastructures considered in a particular scenario.

Considering the complexity of the problem, we have thought about a distributed solution in which all operators can run their own codes offering the provided results coming from them. However, it is not possible a software and operating system normalization because a lot of the available codes are developed in different times (power grid codes, transportation and telecommunication network codes,…). For this reason we have made a solution that leaves every operator offer their own results and simulation codes to high calculus capacity nodes.
The Simulation Center realized an Application Server, which provides different services to final users. All provided services should be given by using both the capacity calculus center and the codes provided by different operators.
The middleware between the Application Server and the simulation codes is made by "wrapper" which interact with the simulation codes in a traditional way and with the Application Server using the Web Service technology with XML for data representation, and HTTP/SOAP for information transport.
In this situation the Application Server can require a node, where a simulation code runs, both for simple data and for the execution of the code with a particular input. This request could be either synchronous or asynchronous: in the first case the protocol sends its request and waits for the results before continuing the elaboration; in the second case it does not attend for the ultimate execution, it goes on its elaboration, and finally asks for the results of the simulation code.

1) A solution proposed

The situation is the following: final users coming from different organizations enter the services provided by the Simulation Center by Internet, and after creating an own account, they should set by the middleware their own simulation requests and send all of them to the Simulation Center, and wait for the final results.

Our purpose was to set up a **simulation platform** where it is possible performing failure and attack scenarios in an automatic way, in order to emphasize problems connected to interdependencies among different complex networks.

CRESCO SPIII.5 structure is built by a system based on a distributed architecture that allows the communication among different tools that runs into the ENEA GRID called **ASSOCIA** (**A**pplication **S**erver for **S**imulation **O**f **C**ritical **I**nfrastructures and data **A**daption).
It consists of two kind of middleware: the first one is a *request middleware* that manages all users requests, and the other one is a middleware between domain simulators and simulation models (*management middleware*). Both have been developed by CRIAI_Consorzio Campano di Ricerca per l'Informatica e l'Automazione Industriale, and they are called **ASSOCIA Web** and **ASSOCIA Core** respectively.

At the moment the **CRESCO SPIII.5** structure has two simulation models (**Ciab** and **Cisia**) developed by Università TOR VERGATA and Università CAMPUS BIOMEDICO respectively. **Ciab** model (**C**ritical **I**nfrastructure **A**gent **B**ased) represents a social emergency scenario, where an Information System for Civic Emergency Management (**IS4CEM**) is developed. **Cisia** model **(C**ritical **I**nfrastructure **S**imulation **by I**nterdependence **A**gents) is a simulation scenario regarding some infrastructures of Nord of Rome zone.

The two simulation models represent two different granularity scenario, and both communicate with a power grid simulator, **e-AGORA**, and telecommunication network simulators, **Omnet** and **NS2** respectively.

In the future it could be possible to add other kinds of domain simulators, such as for example railway networks simulators.

II. CONCLUSION

The objective of this activity was to describe in a formal way the working of a scenario in which different and complex networks work, such as power grid, telecommunication network, transportation network, and so on.., that interact by data information exchange and interdependencies.
The final results coming from simulations should predict networks behavior which are accident-prone by external fault or working state. It could be very important to have a constant analysis of every infrastructures belong to the scenario in order to have a working trend of all the services provided.

ACKNOWLEDGMENTS

The authors would like to thank all the colleagues of the ENEA FIM-CAMO Unit for their supports and contributions. In particular, with respect to all the CRESCO Project participants.

REFERENCES

[1] E.Casalicchio, E.Galli, "Federated Agent Based Modeling and Simulation: an Approach for Complex Critical Systems Analysis", 22nd Workshop on Principles of Advanced and Distributed Simulation, 2008
[2] E.Casalicchio, E.Galli, "Obiettivo 04-CIAB", 2007.
[3] A. Jannace, "CRESCO SPIII.5 ASSOCIA CORE - Il Middleware di ASSOCIA", 2008.
[4] Manuela Aprile, Stefano De Porcellinis, Roberto Setola, Stefano Panzieri, "Simulazione di infrastrutture critiche eterogenee e interdipendenti", TecnicaControllo, 2007.

First Author

- Nationality: Italian
- Birth Date: 04/12/1980

Current Position

- From August 2007: Junior research at the ENEA (Ente per le Nuove tecnologie, l'Energia e l'Ambiente), FIM-CAMO - Computing and Modelling Unit.
- Research areas:
  - System Modeling: modeling and simulation of complex systems.

Education

- October 2006 / June 2007: master's degree "Corso di Perfezionamento in Nuove Tendenze nella Didattica della Matematica e della Fisica" (pedagogy on mathematics and physics) with the maximum score "cum laude" at the the Facolta' di Scienze MM. FF. NN. of the University of Rome Tor Vergata.
- 28/10/2005: degree in **Mathematics** with the score 104/110 at the Facolta' di Scienze MM. FF. NN. of the University of Rome Tor Vergata. The thesis's title was " Il Problema dell'Assegnazione di Potenza in Reti Ad-Hoc: Algoritmi e Teoria Dei Giochi " ("The rating assignation in Ad-Hoc Networks: Algorithms and Game Theory")
- 1999: degree of secondary school with an emphasis on sciences.

VOMS: INFN Contribution to Grid Security

A. Ceccanti[1], V. Ciaschini[1], V. Venturi[1]
[1]*INFN CNAF, Bologna, Italy,*
{*andrea.ceccanti,vincenzo.ciaschini,valerio.venturi*}*@cnaf.infn.it*

***Abstract*—Authorization and Authentication on grid infrastructures are very different problems then their instantiation on a single computer or on a single computing center, due both to the requirement of single sign-on and to the greater scale and dinamicity of the environment. To solve this issue, several concepts have been introduced, in primis that of Virtual Organization (VO) and proxy certificate (Proxy). The main middleware architecture to support, and in some cases create, these ideas has been the Globus Toolkit, which found rapid adoption and constitutes the underlying layer of most grid middleware distributions. Its authorization layer however was excessively simple, since it was based on static user lists that had to be accessible from all head nodes of a grid and, if single sign-on was to be maintained, had to be kept synchronized. Due to this problems, many of its adopters extensively tinkered with this system, rolling out several potentially incompatible extensions.**

INFN has made a fundamental contribution to this field, with the introduction of a tool, called Virtual Organization Membership Service (VOMS), which combined several pre-existing concepts like VOs, Proxies, X.509 Certificates, Attribute Certificates (AC) and Role Based Access Control (RBAC) in an original way, and became a de facto international standard, adopted by all major grids regardless of the specific middleware used (Plain Globus, gLite, UNICORE, Crown, Nordugrid, Naregi, etc...), and thus facilitating interoperable authentication and authorization even among grids that had very little in common with each other. In the following article, we will describe the service, with special attention on the general ideas that shaped it, and on the new abilities that both users and system managers gain from using it and that have driven its wide adoption, along with a few examples of its current usage in the gLite grid middleware.

***Index Terms*—VOMS, Grid, middleware.**

I. Introduction

THE core problem of Grid computing is enabling coordinated resource sharing among dynamic collections of individuals, institutions, and resources, what are referred to as *Virtual Organizations* (VOs) [1]. In a VO, a varying number of participants with various degrees of prior relationships, join in order to share resources. This sharing must necessarily be, highly controlled, with resource providers (RPs) and consumers defining clearly and carefully what is shared, who is allowed to share, and the conditions under which sharing occurs. Resource sharing is conditional: who provides the resources make them available subject to a number of constraints on who is authorized to use them, when, and for what reasons. Such constraints are agreed between RPs and VOs, and Grid middleware need the provide the instruments to enforce such agreements. Authorization plays thus a central role in enabling Virtual Organizations.

An example taken from the environment where VOMS[7] was born is a High Energy Physics experiment. Thousands of physicists at hundreds of laboratories and universities worldwide come together to design, create, operate, and analyze the products of a major detector at CERN, the European high energy physics laboratory. In the experiment (the VO from now on) there are people responsible for collecting the data, who need to be able to write massive amount of data to storage. And there are

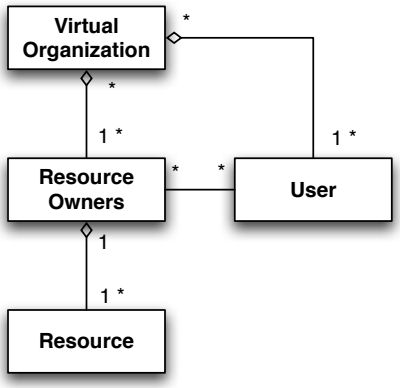


Fig. 1. Relationships in a Virtual Organization

people doing analysis on the data. In a physics experiment, as in any collaboration, there may be conflicting interest, and each organization coming to share in the VO want to be sure the sharing is fair, and it is getting back for what it brings in.

The key to realizing such scenarios is attribute-based access control: authorization mechanisms that allow access control decisions to be made on the basis of a variety of user attributes in addition to simple identity. When combined with a suitable variety of attribute sources, the dynamic authorization needed to enable the sharing needed in a virtual organization become possible.

VOMS was born with enabling VO in mind. It is a source of attributes, that can be used as building block in complex attribute based authorization scenarios. Attributes released by VOMS reflects the organizational relations that exists in VOs. VOMS can state that a user belongs to a certain group in the VO, and that in a certain group possess a role. The VO of the example can enforce authorization policies where user belonging to the production group are allowed to use storage resources during the data collection time, while people in the analysis group are allowed to use computational resource during the analysis data. Or authorization policies that assure that each organization receives resources in proportion to the agreements defined in the VO.

II. VOMS

THE VOMS package may be roughly divided in two different sub-packages, according to their main usage: an Attribute Authority Service (AAS) and a Registration and Administration Service (RAS). The AAS is comprised by a server whose main objective is to release additional attributes to the users, and by the clients which can contact it and create proxies, while the RAS deals with VO management issues: managing users, groups and roles, registering new users, creating grid-mapfiles.

These two sub-packages will be better described below.

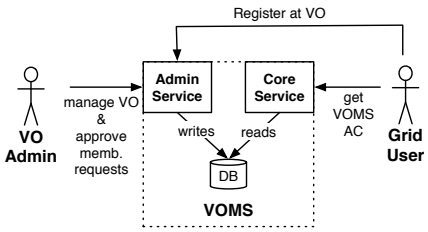


Fig. 2. VOMS Architecture

The main idea of VOMS is to augment user's credentials with additional attributes, thus allowing services to retrieve them and take authorization decisions on their basis.

A. Attribute Authority Services

The main idea of VOMS is to augment user's credentials with additional attributes, thus allowing services to retrieve them and take authorization decisions on their basis. To solve this problem, it issues Attributes in one of two different formats: X.509 Attribute Certificates (AC)[5], and SAML Attribute Assertions (SAML AA)[2].

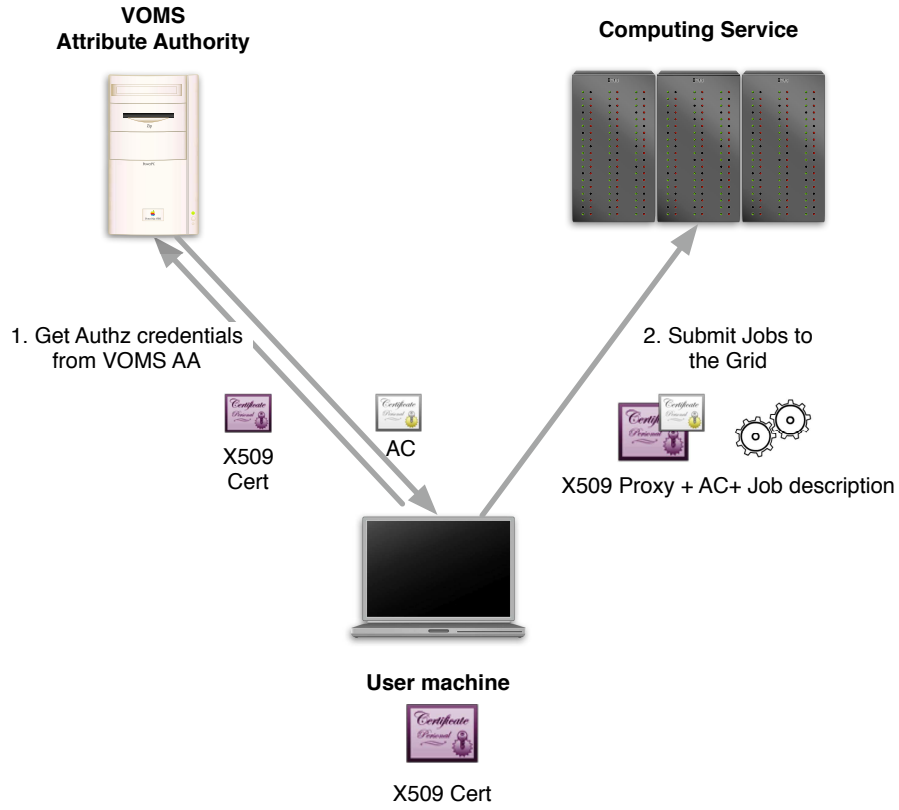


Fig. 3. VOMS Interaction

In the remainder of this section we will concentrate on the AC format, since this is the format primarily used today. Details about the SAML AA will be given in section II-A2.

Now, we have a service capable of creating ACs. This brings about the problem of how to effectively distribute this information to the services. Luckily, users were already used to issuing a `grid-proxy-init` command which created a proxy[6] which would subsequently be sent to the services; it was then a natural extension to also use this proxy to convey the additional attributes.

In order to do this, this service was separated into a client and a server part. The server actually issues the ACs, while the client is used to contact the server and retrieve the AC, provided the user had sufficient privileges to request it.

Multiple vomsd servers belonging to different VOs may be contacted at the same time, and ACs from all of them will be included in the proxy, thus resulting in a full representation of a user's position in multiple VOs, if he is a member of both.

The attributes themselves may be divided into three main categories: groups, roles, and generic attributes.

Groups reflect the organizational infrastructure of the VO, and the user's position inside of it. They are organized in a hierarchical structure, where membership in a subgroup implies membership in a parent group. They are represented in the form of pathnames, where each directory represents a group, and a parent directory represents a parent group. For example, the following: `/dteam/it` represents membership

in the subgroup `it` of the group `dteam`, which is the whole VO. Group membership is always present among the attributes returned in the AC, regardless of the user's wishes.

Roles reflect the user's position inside of a specific group. Not all users are the same, but some may hold special positions. For example, some users called software managers have the additional role of installing software on the machines, which will be used by the other members of the group. Such roles only make sense in the context of a specific group, and thus their representation is closely linked to that of the group. In the example above, the role of software manager in the group of dteam is usually represented thus: `/dteam/Role=sgm`. This intermixing of groups and roles is the main attribute returned by VOMS, and is called Fully Qualified Attribute Name, FQAN for short. Furthermore, the additional abilities granted by roles are not always necessary and thus, in conformity to the principle of minimal privilege, are never returned by default. An user which wishes to receive them, must explicitly request them.

Generic attributes are (name, value) pairs which can be used to represent user attributes that do not directly relate to the organizational structure of the VO (its groups and roles), but identify some other property on which authorization decisions can be taken.

1) Implementation Details: The VOMS Attribute Authority is composed by two elements: a daemon, called vomsd, and a client, `voms-proxy-init`. Each VO has its own daemon, whose task is to authenticate client requests for attributes and issue signed ACs to authorized VO members. `voms-proxy-init` instead is a substitute for `grid-proxy-init`, compatible with it, which is also capable of contacting one or more issues of vomsd and of inserting the returned AC into the proxy.

Additional utilities are also part of VOMS-Core: voms-proxy-info, voms-proxy-destroy and voms-proxy-list.

While voms-proxy-destroy is completely equivalent to grid-proxy-destroy, voms-proxy-info also prints informations from the AC, if present, while voms-proxy-list returns a list of all the attributes that the user may request from the server.

This interaction can also be exemplified by figure 3.

2) SAML Implementation: The Security Assertion Markup Language (SAML) is developed by the Security Services Technical Committee of OASIS. It is an XML-based framework that allows business entities to make assertions regarding the identity, attributes, and entitlements of a subject (an entity that is often a human user) to other entities, such as a partner company or another enterprise application.

SAML defines *Assertions*, packages of information that supply one or more statements by a SAML authority, among which are *Attribute Assertions*. It also defines protocols to request *Assertions* from SAML authorities [2], and bindings into standards messaging or communication protocols [3].

The VOMS SAML[8] service exposes an interface according to SAML protocols [2] and bindings [3], and releases attribute assertions the same way vomsd releases attribute certificate.

B. VOMS Administration and Registration Services

The VOMS Administration and Registration Services are responsible for VO creation, upgrade and management.

VO Administrators use the administration services to define the VO structure (i.e., its groups and roles), assign VO members the appropriate attributes and manage who is allowed to perform perform administrative operations on the VO.

The Administration service implements an authorization framework based on X.509 certificates[4] to allow the execution of VO management operations only to trusted authenticated users. In particular, the frameworks has

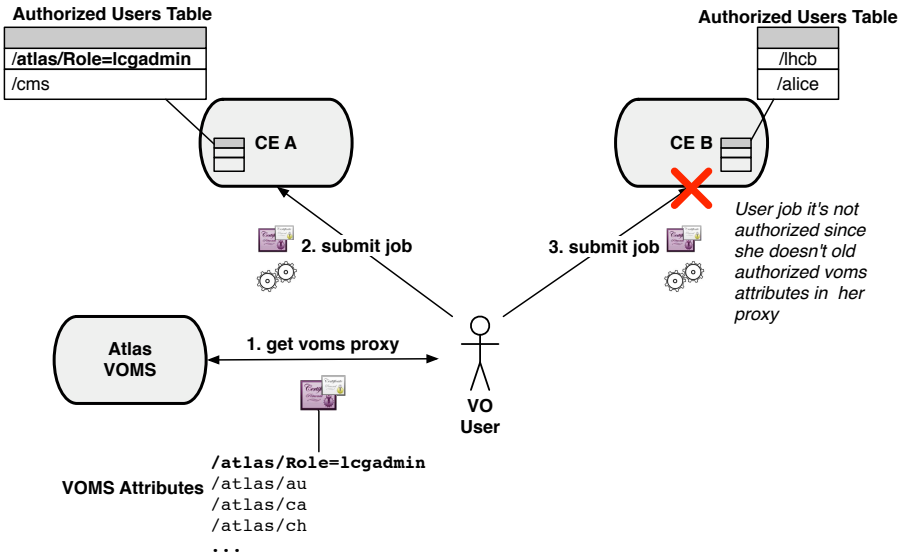


Fig. 4. VOMS Authorization Scenario

Access Control Lists (ACLs) that define the trusted user's capabilities and allow the implementation of delegation of administrative tasks among groups of VO administrators. For instance, it's possible to grant an administrator the possibility to assign membership to users only for a specific group.

All the Administration service management operations are exposed through a Web Service interface, to foster interoperable integration with other applications or middleware components. The web service interface is also the one contacted by the Administrative service command line tools.

The Registration service consists of a web based registration page where aspiring VO members can submit their personal information and submit a VO membership request. Such request is then evaluated by VO managers and if deemed appropriate, the user becomes part of the VO and is assigned VOMS attributes.

1) Implementation Details: The VOMS Administration and registration services are implemented in the Java programming languages as J2EE Web applications. The administration services are accessible either trough a web based interface or using a command line client (`voms-admin`) that enables the scriptability of repetitive administration tasks.

The voms-admin-configure script manages the creation, upgrade and configuration of VO instance.

III. A VOMS USE CASE IN GLITE

SEVERAL High-Energy physics experiments need, to successfully perform data analysis, a rapidly evolving set of scripts and executable, which resource managers cannot really manage. This in turns requires granting a specific subset of users additional privileges to install software on grid resources.

Due to the large number of sites and resources involved, having an explicit list of authorized users that will have to be updated and then redistributed can be a synchronization or coordination problem, due to delayed or mistaken updates. By grouping authorized users with VOMS attributes, we can solve these issues. For example, we may have a "Software-Administrator" VOMS attribute that is assigned to people in the VO that has the responsibility

to install experiment software. Such users can present this attribute to sites which will consequently authorize them to install whatever they need.

This has the advantage that there will never be a need to update any list of allowed users, since all the site need to know is the attribute associated to the install privileges.

Indeed the WLCG deployment used precisely this idea. The various experiments decided to identify a special set of user with the `lcgadmin` role, while sites agreed to authorize users bearing this role to install software.

This scenario can also be seen in figure 4.

This solved the problem: sites did not need to keep up with a changing list of authorized users, while experiments could modify this list at will just by granting or removing this role from users. Additionally, by tieing this ability to a role rather than to a group, they also allowed those same users to also request those privileges when they actually needed them, thus respecting the rule of minimum privilege.

IV. Conclusions

VOMS enables the fine grained authorization needed for sharing resources in a VO. Its attributes allows to model the organizational structure of the VO, making possible the definition and enforcement of authorization policies that reflect the agreements between resource providers and the VO.

VOMS offers a complete set of tools needed to manage a VO. A web application allows users to register, and vo administrators to manage the structure of the VO, while web service interfaces and command line tools allows the automation of administrative tasks.

As described in sectionII-A, VOMS is seamlessly integrated with existing grid tools and procedures, and thus substitutes, adding the capability of augmenting credentials with attributes, the family of grid-proxy-init commands. As such it offers a simple experience for users, that are not requested any change in their procedures, or any configuration but a one time VO registration. VOMS implements the push model for attribute retrieval, where attributes are requested by the user to VOMS and pushed to the resources that will use them to drive authorization decisions. With respect to pull model, where user attributes are requested by resources, this proves to be very scalable and fault tolerant.

References

[1] I. Foster, C. Kesselman, S. Tuecke, *The Anatomy of the Grid: Enabling Scalable Virtual Organizations*, International J. Supercomputer Applications, 15(3), 2001.

[2] OASIS SAML Technical Committee, *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*, OASIS Standards

[3] OASIS SAML Technical Committee, *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0*, OASIS Standards

[4] D. Cooper at al., *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile* Internet Engineering Task Force, RFC 5280

[5] S. Farrell, *An Internet Attribute Certificate Profile for Authorization* Internet Engineering Task Force, RFC 3281

[6] S. Tuecket at al., *Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile*, Internet Engineering Task Force, RFC 3820

[7] R. Alfieri, R. Cecchini, V. Ciaschini, L. dell'Agnello, A. Frohner, A. Gianoli, K. Lorentey, F. Spataro, *VOMS, an Authorization System for Virtual Organizations*, Across Grids 2003, LNCS 2970, pp. 39-40, 2004

[8] V. Venturi, F. Stagni, A. Gianoli, A. Ceccanti, V. Ciaschini, *Virtual Organization Management Across Middleware Boundaries*, eScience 2007, pp. 545-552

Andrea Ceccanti works in the EGEE project since the end of 2005. He is the main developer of the VOMS Administrative service

Vincenzo Ciaschini has been working in Grid Computing since 2002. He was in the DataTag project first, and later in the EGEE project and its successors. He has been from the start, and still is, VOMS's lead developer and designer. He also involved in the design and realization of an XACML-based policy infrastructure for the Grid.

Valerio Venturi has been working with Grid computing since 2004. He was in the EGEE project, contributing to the development and success of VOMS, the authorization middleware adopted by many important Grid middlewares. Subsequently he joined the OMII-Europe project, leading the INFN work on authorization standardization. He's involved in grid standardization bodies, and he has chaired the OGSA Authorization WG of the Open Grid Forum. Since 2008 he has been working in the ETICS project.

Negotiation-Based Advanced Reservation to Manage SLAs in gLite

A. Di Stefano[1], G. Morana[1], G. L. Scuderi[1] and D. Zito[2]
[1]*DIIT, Catania, Italy, {adistefano,gmorana,glscuderi}@diit.unict.it*
[2]*Consorzio COMETA, Catania, Italy, dzito@diit.unict.it*

***Abstract*—The current Grids typically adopt mechanisms and policies about service and task execution, all featured by a characterization of the SLA that was a priori stated for the VO. This is not always satisfactory, since it could not match the dynamic requests from single VO users that might desire at any time a service with a certain profile and according to specific constraints.**
The provision of services with a certain degree of guaranteed parameters implies the existence of tools and mechanisms able to set and monitor the QoS in order to verify that the required constraints are actually supplied. However, the convergence between Grid and web service, combined with the advent of Cloud computing, make the efficient use of resources critical.
This paper propose a software module, Advance Reservation Module (ARM), to address these issues in gLite. The goals of this module are: i) to provide a support for resource reservation and job allocation based on the requirements characterizing the reservation request, ii) to exploit the components of LSF EGO to perform job submission, job execution and state updating on the CE, iii) to introduce the support for immediate allocation and advance reservation on the gLite middleware (but generally enough to be integrated in other middlewares).

***Index Terms*—Advance Reservation, Allocation, Schedulation, gLite.**

I. Introduction

In a scenario where resources and services should be available on demand, the issues related to the management of their complexities will play a crucial role in making the Grid middleware more reliable and pervasive. The grid services have to compete for the resources on the basis of multi-criteria QoS parameters (e.g. cost, capacity, timeframe, utilization, fairness, etc.) facing the middleware complexities, the heterogeneity of the resources, their dynamic behaviour and their membership to different trust domains.

The actual Grid middlewares manage their resources and services with a limited degree of QoS which tipically is obtained adopting algorithm supporting best effort. However the application scenario evolves toward the need to match complex user requirements often strictly related with the opportunity to supply guarantees on the provision of one or more resources. Such guarentees require mechanisms and policies allowing the assignment of a resource to a user or job (or even to a service) for a specific time slot, i.e. making an advanced reservation for one or more resources (e.g. CPUs, bandwidth, storage, etc.). This work mainly focuses on the service of job allocation.

The current Grid infrastructures do not adequately face this issue, resorting often to the classical scheduling policies as FIFO, inherited from HPC. The dynamic Grid behaviour, the under-utilization concerns, the heterogeneous resources and the lack of support for agreement enforcement make advance reservation hard to provide. Presented Grid middlewares handle resource allocation according to the following considerations: (i) deployed Grid middlewares lack support for advance reservations, (ii) applications of interest require the discovery, reservation and allocation of not just a single resource but potentially complex collections;

(iii) resources are commonly located in different administrative domains and subject to different control policies and mechanisms, iv) the provision of advance reservation requires many and complex operations to configure, limit and assure services to a user request. All this operations has to be supplied by the service provider, according to its own for security and accounting policies.

The absence of reservation system, furthermore, gives rise to an additional question: the increase of the costs related with the overprovisioning (as in the Public Switched Telephone Network). On the other hand, if the overprovisioning is understimated, then we econunter degraded service for critical traffic (as in todays Internet). Note that in some systems, such as supercomputers or HPC clusters, overprovisioning may not be an option.

This paper propose a software module, Advance Reservation Module (ARM), to address these issues in gLite. The goals of this module are: i) to provide a support for resource reservation and job allocation based on the requirements characterizing the reservation request, ii) to exploit the components of LSF EGO to perform job submission, job execution and state updating on the CE, iii) to introduce the support for immediate allocation and advance reservation on the gLite middleware (but generally enough to be integrated in other middlewares). ARM uses LSF as Local Resource Manager (LRM) exploiting the components of its Cluster Kernel to perform job submission, job execution and state updating on the CE [19]. ARM enables the construction of application-level co-reservation and co-allocation policies and tools to dynamically assemble collections of resources, guided by both application QoS requirements and local administration policy. Finally ARM is integrated with gLite middleware, even if, for its generalization, it could be integrated into other middlewares.

The paper is organized as follows: II focuses on the description of the issues related with the advance reservation and then discusses the relationship between AR and SLA II-A and gives an implementation pattern II-B. The following section III describes ARM, defining the components III-A and the interactions both with glite II-C and LSF III-B. Finally section IV gives the conclusions.

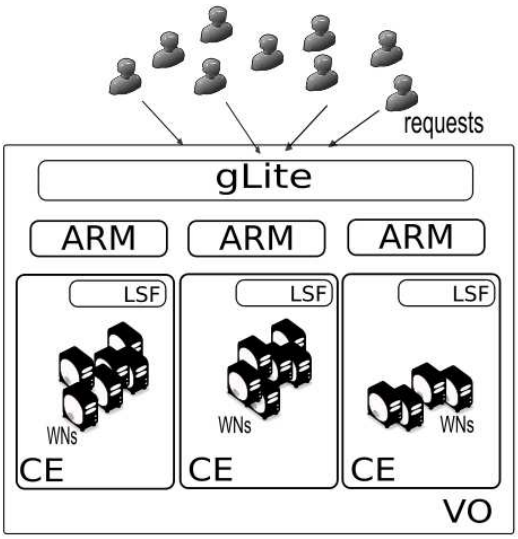


Fig. 1. Architecture at glance

II. QoS and Advance Reservation

A. Issues related to the provision of Advance Reservation in the gLite middleware

Actually, the gLite middleware lacks supports for the management of advance reservation and this issue forces it to make resource allocation at runtime with reduced QoS. Such behaviour strongly reduces the ability of this middleware to follow adequately the user requirements. In a purely *best effort* logic, the middleware cannot influence the provision of the service, because it has not tools to guarantee the resources required for an optimal supply of the service. This guarantee can be provided only through an advance reservation, i.e. *a possibly limited or restricted delegation of a particular resource capability over a defined time interval, obtained bay the requester from the resource owner through a negotiation process* [15]. Naturally, the integration

of AR in a grid middleware poses new issues to face; it occurs, in fact:

- to limit resource utilization according to users requirements
- to find an allocable time interval in order to provide a time fesible scheduling for each resource
- to discovery the available resources
- to combine the reservations with the existent allocations for a given resource
- to handle the user's error in the walltime prediction. If the user overestimates the walltime for its service execution, there will be underutilization of resource; while if the user underestimates the walltime, it occurs to estabilish the roles through wich to assign new resources to the reservation.

Besides the above logical problem there are many other integration issues, more related with the middleware implementation that makes complex this integration (for more considerations about these aspects see III-B and II-C).
Furthermore, the advance reservation is strictly related to the allocation and the scheduling of the resources. The networking research literature frequently uses the terminology *immediate reservation* to indicate the simple allocation (e.g., the ReSerVation Protocol: RSVP [16]). This definition remarks the fact that an allocation is a subcase of the classical advance reservation, where the starttime is not indicated and is posed to *now*, when the allocation starts. The central issue is to make possible the coexistence of allocated services with the reserved ones, choosing opportunely the policies and the action strategies. In particular, the task of resource allocation implies i) the resource selection, i.e. discovering and matching nodes to each service (or part of it), ii) finding of allocatable time intervals on each node during which a service or a subservice can execute, and finally iii) match between the available time intervals and the available nodes.
An important consequence of the presence of an AR management is the possibility to introduce the opportunity of providing guaranteed services in the gLite middleware. One of the cases more frequent in the Grid utilization is the request of allocation of resources to execute one or more services of job allocation. In this context, the *best effort* strategy allow the users only to request a fixed amount of resources, but it does not provide any guarantees that each resource will be utilized in a given time range or that will start to be utilized at a fixed time. For more users, this strategy is too restrictive, because the execution of particular classes of services strongly dependend by time constraints. In the field of e-science there is a large set of potential applications that would take advantages by the possibility to specify QoS parameters in their request, especially the time-critical service that requires specific guarantees about their execution. Generally these services provides prediction (e.g. volcanic activity prediction), simulation (e.g. earthquake simulation, ecosystem simulations)and modeling (financial modeling, climate/weather modeling). Such typologies of services are very expensive in terms of computation requirements and can be satisfied by the high computational capabilities of the middleware Grid. However, some application are mission critical, in that their results have to be provided within specific time range, and the violation of these issue are potentially dangerous for the users. The absence of a support for the management of such QoS parameters strongly limits the possibility for these time critical applications to use the gLite middleware. The only way to guarantee the respect of a time constraint is to reserve a resource (or a set of resource) and is strictly related to the issues of QoS negotiation and advance reservation. The opportunity to offer service with a specific QoS profile depends on the possibility to perform specific reservations at CE tier.
Obviously, all the issues related to the QoS provision can not be reduced only to the question of advanced reservation, but it needs adequate strategies to manage several QoS requests.
When the guarantees on the required QoS can be provided, user can establish an agreement that matches his own requests to service of-

fers by a service quality level negotiation. The related result is called Service Level Agreement (SLA), a contract that contains all negotiation parameters. Thus the service provider, according to SLA, configures and reserves its resources to obey negotiated parameters.

B. Implementation Pattern

There are many ways through which a reservation can be performed. E.g. it is possible to reserve a resource for the execution of a specific service, then the reservation ends when the service terminate its execution. On the other hand, it is possible also to reserve a set of resources for a user rather than a service. In this way, when the execution of a service terminates, the resources result still reserved to execute a further service on behalf of the user. The two options are both valid.

The choose between the above options must be functional to the choices implemented in the definition of the SLA drawn up for a specific service. For this reason we defined an *implementation pattern* that is able to follow the tipologies of SLA (as described in previous papers []).[1] In particular:

- *The AR must be defined for a job* In this manner we have one SLA for each specific request. The alternative, adopted e.g. in LSF, is to associate the reservation with an user, i.e. the user reserves a resource and then it can allocate a set of job on it. According to this solution, however, it's difficult to differentiate the SLA for every the jobs/services that belongs to an user and, moreover, every alteration on a SLA would have consequences on every service that the reservation refers, making too complex the renegotiation mechanism.
- *The resource has to be released once the job is completed* This property is loosely connected with the previous one: if an AR is associated with a service then it's necessary to release the resources involved in the reservation once the job is completed, so as to avoid wastes and overages.
- *An active reservation can be altered.* This feature is crucial for the renegotiation of SLA previously drawn up. In fact, if the user starts a process of modification of the parameters fixed in a SLA, this process will be completed if and only if this alteration is visible on the low level LRMs.
- *A reservation can be negotiated.* The alteration of a reservation at run time, in consequence of a political choice at high level, could be negotiated if the desired resources have different service levels or the same service levels but with different costs. This renegotiation should involve the two parties that participate to the SLA.
- *Reserving portions of resources.* The possibility to reserve portions of resources, instead of whole resources, allows the users a more flexible use of them and, above all, enables the implementation of interesting scheduling policies and load balancing strategies to avoid, where it is possible, wastes and overage. This property is loosely connected with the issue related to allocation and scheduling that are not considered in the present paper.
- *Defined and undefined Reservations.* The defined reservations are the AR with both start time and end time, while the undefined ones are the AR that have only the start time. Obviously it's more hard to face the undefined than defined reservations, because of the lack of a specified end time.

To make a reservation, the reservation manager first must perform a scheduling simulation of applications currently in the system (basing on the monitoring information) and then produces a timeline of when nodes will be used in the future. This timeline is then used to determine *if* and *when* a reservation for an application can be made. The reservation manager uses an estimation of maximum run times (see [18]) when creating the timeline. This provides more assurances that the reserved applications do not conflict with the running applications or other

[1]For the status of Advance Reservation we refers to: http://www.fz-juelich.de/jsc/RD/coop/ggf/graap/sched-graap-2.0.html#Sched

reserved applications.
Each conflict for a processing interval implies, in fact, the termination of the running allocated applications (if they are not reserved), because the reservations have to be honoured.

C. gLite Integration

GLite is the standard middleware proposed by EGEE; it is based on EDG experience and it integrates the best Grid project component like Condor [10], MDS Globus [12], LCG [11], and VDT [9]. GLite offer an high level services set to manage computational resource, scheduling, storage operation, resource and application information. These technologies enable an easy interaction between services allowing interoperability among different components. All these features promote this architecture as an upcoming Grid standard. ARM iteracts with two main gLite components: the Relational Grid Monitoring Architecture (R-GMA [7]) and the Workload Manager Service (WMS [8]). The first one, R-GMA, is an implementation of the Grid Monitoring Architecture (GMA) proposed by the Global Grid Forum (GGF), that provides a service for information, monitoring and logging in a Grid environment. This component abstracts the complexity related to the resources information management, virtualizing and collecting them in one large Relational Database that may be queried directly by the users via web services. The main feature of this architecture is the ability to provide a useful and predictable information system built on the data provided by losser-coupled providers across the grid environment. The R-GMA architecture consists of three components: i) *Producers* which publish information into R-GMA, *Consumers* which obtain the information by subscribing to the interested services and *Registry* which mediates the communication between the Producers and the Consumers offering them a common interface.
ARM interacts with the R-GMA to retrieve and integrate the information about the allocations presents on every CE. These information updates the PlanningTable (the information index of ARM) about the state of the CE. From the R-GMA perspective, the advanced reservations are completely transparent, it only sees a list of allocations on a set of resources. It is responsability of the ARM, to implement the advance reservation on the resources and to notify the R-GMA about the status changes.
On the other hand, WMS is the component that handles the jobs allocation issues. It works on i) accepting requests for job submission and management coming from its clients and ii) taking the appropriate actions to satisfy them. ARM integrates the role of the WMS, adding the functionalities related to the management of reservation requests, that are not present in gLite.

III. ARM

The Advanced Reservation Manager (ARM), is the component of the proposed architecture used to opportunely manage the LRMs of the CE, allowing the translation of the reservation requests to actions directly executed on the resources. It acts at the CE tier, i.e. there is an ARM in each CE, and interacts with the underlying LRM.
The goals of this component are:

1) to provide a support for resource reservation and job allocation based on the requirements, the constraints and the parameters characterizing the reservation request;
2) to offer an interface that makes possible an efficient use of resources;
3) to introduce the support for immediate and advanced reservation on the gLite middleware. item to exploit the components of platform EGO to perform respectively job submission, job execution and state updating on behalf of the reservation service;

A. Components of ARM

ARM consists of five main components:

- **ARMServer** waits for users reservation requests. For each received request, ARMServer checks the Job Descriptio Language

(JDL[2]) file to verify whether reservation parameters are properly defined (e.g. they belong to a correct range) and whether the user has the authorization to request the specified type and the given amount of resource. If the checks succeed, ARMServer forwards the request to ARMController and sends the reply to the user.

- **ARMController** performs all the activities related to the resource reservation. It checks whether a suitable resource is available, asking to the TableManager for potential free slots on the planning table. ARMController starts, monitors and terminates: i) the allocation (through the JAM), the advance reservation and the scheduling of jobs to the target resources. When the start time of an advance reservation is reached, ARMController drives JAM to submits the job to the reserved resource. Futhermore, ARMController deletes served or inactive requests and records all of its activities on a log file. ARMController also has a local fault recovery engine that, in case of faults (e.g. the WN is unreachable because it has crashed before the start time), it is able to restore, where possible, the correct job execution process.
- **JAM** is the sub-component of ARM that deals with the issues related to the jobs allocation. It consists of three main thread that directly interact with the EGO cluster kernel.
- **TableManager**is an interface between the planning table and the rest of ARM. It handles the planning table, updating its entries with the information of the EGO cluster Kernel and interacting with the ARMController every time that a reservation is required. It is responsible for the integrity of the data held in the planning table and it is the only component able to add, remove or modify a entry of the planning table.
- **Planning table** is the information source of ARM: it integrates the information held in the EGO Cluster Kernel Information Service with the information about the active reservations within the CE. It is a key component for the reservation mechanism because, in addition to the information about the actual status of the CE (needed to perform the simple allocation), the planning table contains some indications about the future status of the resources (in terms of reservation done on it in advance). This information has a fundamental importance because it makes possible to establish whether a reservation could or could not be accepted. The planning table is constituted by a hash table that holds all the resources managed by the CE. Each resource, furthermore, maintains a specific hash table with the start time as the primary key and the end time as the secondary key. In this way, the TableManager first queries for the resources and afterwards verifies, through start time and end time, if there would be potential overlaps. It returns this information to the ARMController which, based on the defined reservation policies, will take the most suitable decision.

[2]a high-level, user-oriented language based on Condor classads for describing jobs and aggregates of jobs such as Direct Acyclic Graphs (DAG) and Collections.

B. Integration with LSF

In our work, Platform LSF is the LRM used as a low level tool to build an architecture for the management of advance reservation according to the SLAM choices. In the version 7.0 of LSF, it was introduced the Platform Enterprise Grid Orchestrator (EGO), a service that virtualizes a distributed heterogeneous computing environment providing the final users with the vision of a collection of distributed software and hardware resources as components of a single virtual computer. It determines the number of resources each consumer is entitled to use, takes into account a consumers priority and overall objectives and then allocates the number of required resources (for example, the number of slots, virtual machines or physical machines).

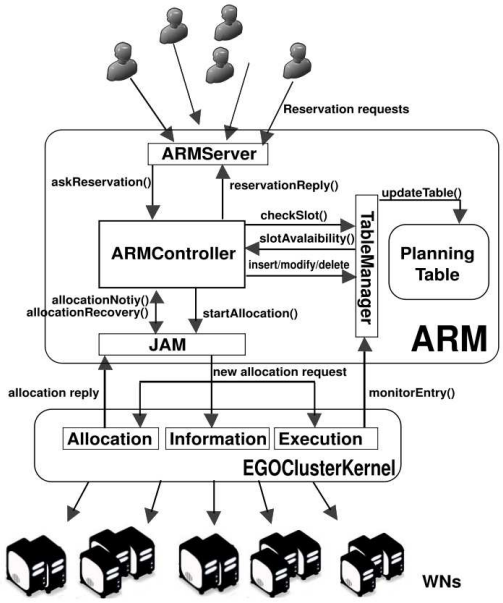


Fig. 2. ARM

The Platform EGO is used to abstract the CE, hiding the complexity of the management of its resources from the architecture implemented. The architecture of EGO comprises a collection of software components, even if the core functionalities (allocation, information and execution) are grouped in the Cluster Kernel (vemkd). These functionalities are: i) *Allocation Service*: which manages allocation requests coupling resources requests with resources availability by clients; ii) *Execution Service*: which manages execution states flow, by checking each process status offering a primitive to stard and stop job; iii) *Information Service*: it is the resource and job monitor that collect information on each consumer, resource and allocation information. Although this LRM is a widely consolidated product in the Grid environment, it lacks mechanisms and tools to allow the management of the advanced reservation profile explained above. In order to overcome these limitations, a new software module has been implemented, operating at the CE tier, able to interact directly with the LSF to obtain the AR.

This module, named Job Allocation Manager (JAM), uses the Platform EGO: i) to have a uniform vision of the available resources in the considered CE and ii) to submit a job maintaining the information about its state until its completation. JAM, as said before, is driven by the ARMController, the core of the whole ARM. JAM is not able to distinguish between an allocation request and an advanced reservation request in that it considers the allocation a border line reservation with start time set up on the current time. The different management of both typologies of requests (allocation and reservation) is demanded by the ARMController.

When the ARMController forwards the allocation requests to the JAM, the requests are queued in the specific resource queue. The JAM is responsible for getting the allocation request from the resource queue and making an allocation request to Platform EGO. The Cluster Kernel processes the requests and sends a reply to the JAM. These replies are held in the work queue. Afterward JAM gets the allocation reply from the work queue, adds the resource to the resource collection structure and starts a container on the allocated host slot.

JAM is also responsible to maintain the information about the status of the resources, properly querying to the underlying information system provided by Platform EGO Cluster Kernel.

IV. Conclusion

In a Grid environment, characterized by a high level of resources heterogeneity and relationship dynamism among participants, the resources could quickly change in an unpredictable manner. Guaranteeing specific service levels in this scenario is very complex. For an advanced reservation management that is able to guarantee a specific resource with an adequate service level for a specific walltime, it is necessary to implement proper mechanisms. In this paper, resource reservation has been considered as a strategy to guarantee that services will give what needed at the proper time.

We discussed the relationship between the SLA and the AR, proposing an implementation pattern able to follows several choices related to the stipulation of a SLA. In according with these concepts, we exposed a software module, ARM, to address the issues related with the sceduling, allocation and advance reservation of services/resources on a CE.

We take care particulary the question of integration with the existent gLite middleware, describing the interaction among ARM and the components of gLite (at application tier) and LSF (at CE tier).

REFERENCES

[1] ITU-T (CCITT),Recommendation, *Terms and Definitions Related to The Quality of Service and Network Performance including Dependability*, August, 1994

[2] A. Bhargava and B. Bhargava, *Measurements and Quality of Service Issues in Electronic Commerce Software*, In Proceedings of International Conference on Application-Specific Software Engineering and Technology (ASSET), Texas, March 24-27 1999, IEEE.

[3] J. A. Zinky, D. E. Bakken, and R. D. Schantz. *Architectural Support for Quality of Service for CORBA Objects. Theory and Practice of Object Systems*, January 1997.

[4] G. von Bochmann and A. Hafid, *Some principles for quality of service management*, In Distributed Systems Engineering, vol 4., numb. 1, 1997.

[5] EGEE, *gLite HomePage*,http://glite.web.cern.ch/glite/

[6] L. Field, *Berkeley Database Information Index*, Available on: https://twiki.cern.ch/twiki//bin/view/EGEE/BDII

[7] EGEE project, *R-GMA development page*, http://hepunx.rl.ac.uk/egee/jra1-uk/r-gma/

[8] EGEE, *EGEE HomePage*,http://www.eu-egee.org/

[9] VDT, *homepage*,http://vdt.cs.wisc.edu/index.html

[10] J Frey and T Tannenbaum and M Livny and I. Foster and S. Tuecke, *Condor-G: A Computation Management Agent for Multi-Institutional Grids*, Cluster Computing, 2002,num. 3, vol. 5,pp. 237–246

[11] LCG project, *official website*, http://lcg.web.cern.ch/LCG/

[12] X. Zhang and J. M. Schopf, *Performance Analysis of the Globus Toolkit Monitoring and Discovery Service, MDS2*, CoRR, 2004.

[13] J. Hathaway, *Service level agreements: keeping a rein expectations*, in SIGUCCS, DBLP, http://dblp.uni-trier.de/db/conf/siguccs/siguccs1995.html, Hathaway95, USA:ACM, 1995.

[14] L Jin and V Machiraju and A. Sahai, *Analysis on Service Level Agreement of Web Services*, In Hewlett Packard Laboratories (HPL-2002-180), jul 2002.

[15] J. MacLaren, *Advance reservations: State of the Art*, GGF GRAAP-WG, August, 2004, www.mcs.anl.gov/ schopf/ggf-sched/GGF5/sched-GRAAP.2.pdf

[16] R Braden and L. Zhang and S. Berson and S. Herzog and S. Jamin, *Resource reservation protocol: RSVP*, sep, 1997, ftp://ftp.internic.net/rfc/rfc2205.txt

[17] W Smith and I Foster and V. Taylor, *Scheduling with Advanced Reservations*, IPPS: 14th International Parallel Processing Symposium, IEEE Computer Society Press, 2000

[18] W. Smith and I. Foster and V. Taylor, Predicting Application Run Times Using Historical Information, The 4th Workshop on Job Scheduling Strategies for Parallel Processing, 1998

[19] F. Gagliardi and B. Jones and M. Reale and S. Burke, *European DataGrid Project: Experiences of Deploying a Large Scale Testbed for E-science Applications*, Lecture Notes in Computer Science, vol. 2459, 2002

A QoS-aware Distributed Discovery Algorithm for P2P-Grid Systems

A. Di Stefano[1], G. Morana[1] and D. Zito[1,2]
[1]*DIIT, Catania, Italy,{adistefano,gmorana,dzito}@diit.unict.it*
[2]*Consorzio COMETA, Catania, Italy*

***Abstract*—The Grid computing paradigm, born in the mid 1990's to enable the sharing of computational and storage resources among academic and research institutions, has been deeply influenced in recent years by the Service Oriented Architecture (SOA) advent.**
The SOA technologies, in fact, allow Grids to open towards new applications, not only in the field of e-science, but also in the business, financial and educational ones.
The Grids, today, are able to supply resources and services in a flexible manner, exposing and offering them on-demand to different typologies of users, each one characterized by specific requirements. It is creating a marketplace for competitive Grid services where the user's satisfaction of the QoS requirement becomes the fundamental issue. In this paper the authors illustrate an effective QoS-aware algorithm to search services and resources in this new distributed scenario and propose a gLite component to make the middleware able to support the proposed discovery mechanism.

***Index Terms*—Peer to Peer, Grid, Discovery Protocol, QoS**

I. Introduction

THE adoption and the use of Service Oriented Architectures (SOAs) into middlewares, with their great flexibility and the adoption of the open and standard Web Services technologies, has deeply changed the way the users can interact with the Grids [1], [2]. Today, the Grids are opening towards new applications, embracing not only the e-science field, where they are born and developed, but also the business, financial and educational ones.

The Grids have now to be able to supply on-demand resources and services in a flexible manner, exposing and offering them to different typologies of users, each one characterized by specific requirements.

The current target of various Grid projects is to create a marketplace for competitive Grid services where the user's satisfaction of the QoS requirements becomes the fundamental issue. In fact, this Grid evolution offers a great number of Grid services that the users can easily access through the web. It can choose and use those services that better satisfy these requirements, obtaining them by different Grid providers basing on the services performance and cost.

In this distributed scenario, characterized by the lack of a centralized coordinator, the user has to rely on a robust and scalable algorithm to find services under QoS constraints among the different providers.

The services, in fact, are distributed among several remote providers, i.e among several Grids: in order to find the providers of a specific service, the consumer has to refer to a services information index.

The design of these services is an important aspect because it has a great impact on the system as a whole.

If a centralized (or hierarchical) services index is used, as with classical Grid solutions (e.g. MDS for Globus Toolkit [23] [25] and BDII for gLite [8]), it is possible to obtain a discovery mechanism simple to manage and characterized by a low response time (i.e. the time needed to obtain the service reference). Unfortunately,

this approach compromises both the scalability and the reliability of the services information index, especially in terms of robustness and fault tolerance as it represents a *single point* of failure.

If, instead, a structured distributed information mechanism is used, like Distributed Hash Tables (DHT [14], [15], [19], [16], [20], [17], [18]), it is possible, exploiting their well-defined structure and their self-organization capabilities, to obtain a low response time and to solve the problems related to scalability and fault tolerance. Unfortunately, this information organization foresees that the information related to a provider and to its offered services could be maintained by another provider: it could be a problem in a system where each service provider is in competition with the others. Using DHT, in fact, it is possible that the information related to several providers of a specific service are stored in another provider of the same service: thus there are no guarantees that this provider responds to a query giving the users only its own service reference, excluding other providers.

To overcome these drawbacks, the authors have proposed in [26] an unstructured distributed services discovery based on a selective flooding algorithm. The proposed algorithm propagates the searching messages through the "near nodes" of the underlying overlay network exploiting a *ad hoc* flooding based algorithm. In this work, the authors briefly discuss the proposed algorithm (Section 2) and explain how it has been implemented and deployed in the gLite middleware (Section 3).

II. The proposed algorithm

The algorithm proposed in this paper has the aim to provide an effective mechanism to find services and resources in a distributed *P2P grid*([21], [22], [24]) based services marketplace. This environment, shown in fig. 1, consists in a distributed overlay network where each peer is a Computing Elements (CEs [13]). Each CE manages directly its underlying Worker Nodes (WNs): in this P2P schema, the concepts of

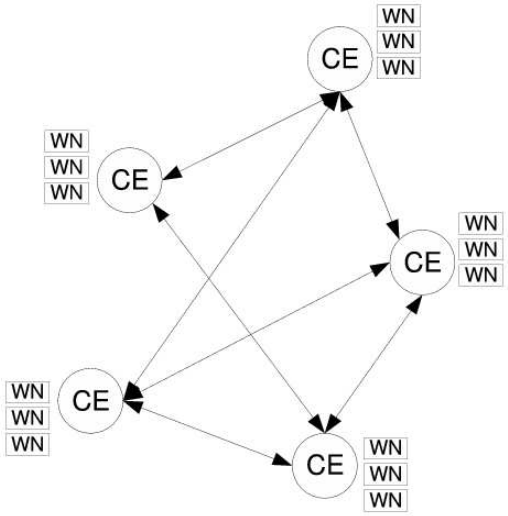


Fig. 1. P2PGrid based services marketplace

centralized Resources Broker and User Interface are missing in that each peer is able to require and provide services to other peers.

The proposed algorithm is explained in depth in [26]: here, a shortly description is given.

When an user wants to exploit a specific service, he has to identify, among the available peers, the ones that provide it.

In order to perform this task, he forwards to a subset of its neighbors, maintained in a *neighbors list* and chosen basing on a *reputation* value, a Discovery Query (DQ) frame.

This frame contains all the information related to the required service, including a list of the QoS([3], [4], [5], [6]) parameters that a generic provider has to be able to satisfy. The ability to search services under QoS constraints, that is an important innovation in the field of discovery algorithm for distributed systems, represents a fundamental value-added in a service marketplace scenario in that it allows not only to identify different providers for the desired service but also to discern among them the most suitable one to satisfy the user requirements.

When a neighbor receives the DQ, searches the services among the ones locally provided.
If it is a provider of the required services and is able to satisfy all the QoS requirement contined in the DQ, it sends back to the user, that has required the service, a Discovery Query Hit (DQH) frame. This frame will contain a proposal for the service execution based on a Software Level Agreement(SLA [11], [12]).
If, instead, the peer that has received the DQ is not able to provide the required service, it queries a local cache, called *services cache*, where it maintans the information about the services provided by other peers. In case of success, it sends a Service Indication (SI) frame to the user to inform him that there is a peer able to provide the required service. This mechanism represent an fundamental solution to strongly reduce the number of frames spread on the overlay network, mitigating one of the drawbacks of the flooding base discovery algorithms. In case that the neighbor does not find the service neither among local provided services nor in its cache, it decreases the hops field of the received DQ frame and, if the hops values is bigger than 1, forward to its neighbors. This process ends when the hops value becomes 1.
The user receives all the DQH (and SI) frames sent by the other peers and chooses, among the available ones, the best provider basing on the fulfilment of the QoS requirements.

III. The algorithm in gLite

Currently, the gLite middleware does not offer any software components able to support the proposed discovery algorithm. gLite, in fact, has been designed for the classical grids and it is unable to manage inter-grid communications, a fundamental requirement for the considered P2PGrid scenario. In order to overcome these restrictions, we have designed and developed P2PProxy,an *ad hoc* component able to manage the messages exchanged in the proposed algorithm decoupling the frames reception and analysis phases from the services execution one. This allows to hide the mechanisms needed for the P2P communications making gLite unaware

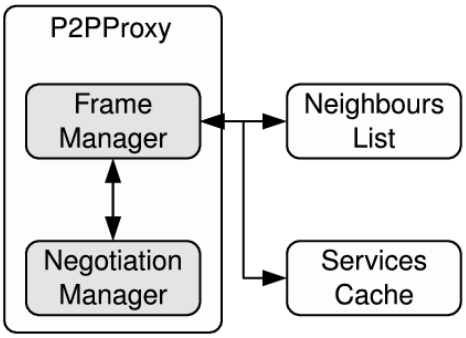


Fig. 2. P2PProxy, neighbors list and services cache

to be a peer of an overlay network.
The main aims of P2PProxy are:

- support and manage all the frames foressen by the discovery algorithm, both input and output ones;
- support the negotiation of functional and non functional(i.e. QoS) parameters and the creation of the related Software Level Agreement (SLA);
- maintain the neighbors list;
- maintain the services cache;

From the architectural point of view, the P2PProxy component has been divided in two different sub-components:

- the frameManager (P2P-side)
- the negotiationManager (Grid-side)

The figure 2 shows the architecture of P2PProxy component, including both neighbors list and services cache that, although tey do not belong directly to P2PProxy, play a fundamental role in the discovery algorithm. The frameManager represents the P2P interface of the middleware: its aims are to receive/send[1] the frames from/to the overlay network and to analyze them in order to understand which type of action has to be taken. The core of the framework is represented by the frameAnalyzer: this component analyzes the each single field of the received frames and, basing on their content, invokes a specific method to manage them.

[1]The component has to be bidirectional.

Since the algorithm can provide information related both the local provided and the known remote available services, the frameManager has to also interact, respectively, with the underlying information index service (in gLite BDII [8] and/or RGMA [9]) and with the services cache. In the first case, it exploits the services offered by the ResourcesFinder, a subcomponent of the SLAM [27] framework that provides a common and uniform way to query the middleware information index. In the second case, instead, the frameManager queries the services cache directly, using a local lookup method.

The negotiationManager, instead, is the component that manage the negotiation and SLA creation phases and comes into action when the frameManager receives or sends the DQH, NM, SA or SC frames. The aim of this component is to verify both the ability of the underlying system to satisfy the specific users requirements contained in the DQ frames (if the peer has the role of provider) and to evaluate the agreement proposal received from remote providers (if the peer acts as service consumer). In order to perform these tasks, the negotiationManager interacts with the SLAM [27] (Software Lever Agreement Manager) framework, a previously developed gLite component able to introduces in this middleware the ability to create and manage the SLAs. The figure 3 shows the interactions between the P2PProxy and SLAM components.

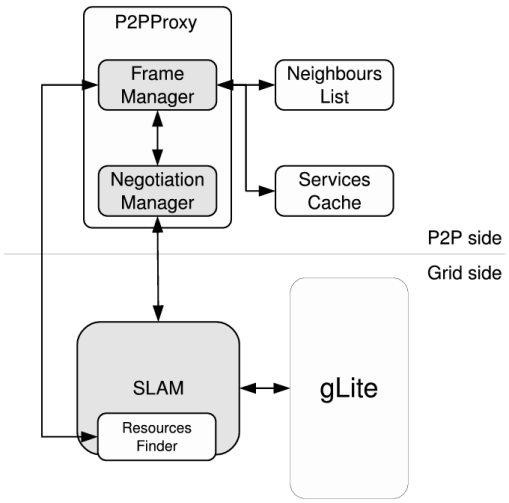


Fig. 3. The interaction between P2PProxy and SLAM

IV. Conclusion

In this paper, the adoption of an unstructured distributed QoS-aware services discovery algorithm and its implementation and deployment on gLite middleware have been discussed. The aim of the proposed algorithm is to provide an effective mechanism to find services and resources in a distributed P2P grid based services marketplace. One of the most important aspects of the algorithm, discussed in depth in [26], is the ability to search services under QoS constraints, that is an important innovation in the field of discovery algorithm for distributed systems: this ability represents a fundamental value-added in a service marketplace scenario in that it allows not only to identify different providers for the desired service but also to discern among them the most suitable one to satisfy the user requirements. The paper has also shown the architecture of P2PProxy, an *ad hoc* gLite component designed and developed in order to make the considered middleware able to support the proposed solution for the distributed discovery service.

References

[1] Congduc Pham, Bernard Tourancheau, Grid infrastructures: practice and perspectives, Future Generation Computer Systems, Volume 21, Issue 2, Advanced Grid Technologies, 1 February 2005, Pages 247-248, ISSN 0167-739X, DOI: 10.1016/j.future.2003.10.001.

[2] Giovanni Aloisio, Domenico Talia, Grid Computing: Towards a New Computing Infrastructure, Future Generation Computer Systems, Volume 18, Issue 8, October 2002, Pages v-vi, ISSN 0167-739X, DOI: 10.1016/S0167-739X(02)00079-1.

[3] ITU-T (CCITT),Recommendation, *Terms and Definitions Related to The Quality of Service and Network Performance including Dependability*, August, 1994

[4] A. Bhargava and B. Bhargava, *Measurements and Quality of Service Issues in Electronic Commerce Software*, In Proceedings of International Conference on Application-Specific Software Engineering and Technology (ASSET), Texas, March 24-27 1999, IEEE.

[5] J. A. Zinky, D. E. Bakken, and R. D. Schantz. *Architectural Support for Quality of Service for CORBA Objects. Theory and Practice of Object Systems*, January 1997.

[6] G. von Bochmann and A. Hafid, *Some principles for quality of service management*, In Distributed Systems Engineering, vol 4., numb. 1, 1997.

[7] EGEE, *gLite Home-Page*,http://glite.web.cern.ch/glite/

[8] L. Field, *Berkeley Database Information Index*, Available on: https://twiki.cern.ch/twiki//bin/view/EGEE/BDII

[9] EGEE project, *R-GMA development page*, http://hepunx.rl.ac.uk/egee/jra1-uk/r-gma/

[10] EGEE, *EGEE Home-Page*,http://www.eu-egee.org/

[11] J. Hathaway, *Service level agreements: keeping a rein expectations*, in SIGUCCS, DBLP, http://dblp.uni-trier.de/db/conf/siguccs/siguccs1995.html, Hathaway95, USA:ACM, 1995.

[12] L Jin and V Machiraju and A. Sahai, *Analysis on Service Level Agreement of Web Services*, In Hewlett Packard Laboratories (HPL-2002-180), jul 2002.

[13] F. Gagliardi and B. Jones and M. Reale and S. Burke, *European DataGrid Project: Experiences of Deploying a Large Scale Testbed for E-science Applications*, Lecture Notes in Computer Science, vol. 2459, 2002

[14] Adeep S. Cheema and Moosa Muhammad and Indranil Gupta, Peer-to-Peer Discovery of Computational Resources for Grid Applications, in SC'05: Proc. The 6th IEEE/ACM International Workshop on Grid Computing, pages:179-185, IEEE/ACM, Seattle, Washington, USA, nov, 2005

[15] Min Cai and Martin R. Frank and Jinbo Chen and Pedro and A. Szekely, Maan: A Multi-Attribute Addressable Network for Grid Information Services",in 4th International Workshop on Grid Computing (GRID 2003), 17 November 2003, Phoenix, AZ, USA, Proceedings, IEEE Computer Society, 2003, pp. 184-191

[16] Ashwin R. Bharambe and Mukesh Agrawal and Srinivasan Seshan: Mercury: supporting scalable multi-attribute range queries,in Proceedings of the ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August, 2004, Portland, Oregon, USA, ACM 2004, pp. 353-366

[17] Maymounkov and Mazieres. Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. In proceedings of International Workshop on Peer-to-Peer Systems 2002, (IPTPS), LNCS",

[18] KaZaA official website http://www.kazaa.com/us/index.htm

[19] Jeannie R. Albrecht and David L. Oppenheimer and Amin Vahdat and David A. Patterson, Design and implementation trade-offs for wide-area resource discovery,in ACM Trans. Internet Techn, number 4, volume 8, 2006

[20] Rajiv Ranjan and Aaron Harwood and Rajkumar Buyya, Peer-to-peer-based resource discovery in global grids: a tutorial, Communications Surveys & Tutorials, IEEE , vol.10, no.2, pp.6-33, Second Quarter 2008

[21] D. Talia and P. Trunfio. Adapting a pure decentralized peer-to-peer protocol for grid services invocation. Parallel Processing Letters, World Scientific Publishing Company, Jun 2005, Vol.15, num.2, pp. 67-84.

[22] J. Tang, M. Zhang, R. Buyya, and T. Ma. An Agent-based Peer-to-Peer Grid Computing Architecture - Convergence of Grid and Peer-to-Peer Computing. The proceedings of the Fourth Australasian Symposium on Grid Computing and e-Research (AusGrid 2006) and the Fourth Australasian Information Security Workshop(Network Security) (AISW 2006), Hobart, Tasmania, Australia, January 2006. ACS, Hobart,Vol. 54, Feb 2006, Australia, pp. 33-39.

[23] I. Foster. Globus toolkit version 4: Software for service-oriented systems. In Journal of Computer Science and Technology, Springer Boston, Volume 21, Number 4, July 2006, pp. 513-520.

[24] Foster and Iamnitchi. On death, taxes, and the convergence of peer-to-peer and grid computing. In 2nd International Workshop on Peer-to-Peer Systems (IPTPS'03), LNCS, February 2003, Berkeley, CA.

[25] X. Zhang and J. M. Schopf. Performance Analysis of the Globus Toolkit Monitoring and Discovery Service, MDS2. In Proceedings of the International Workshop on Middleware Performance (MP 2004), part of the 23rd International Performance Computing and Communications Workshop (IPCCC), Phoenix, USA, April 2004.

[26] Di Stefano Antonella, Morana Giovanni, Zito Daniele. QoS Aware Services-Discovery in a P2P Grid Environment, International Conference on Pervasive Computing and Applications (ICPCA07) - Birmingham, UK - 2007, July, 26-27

[27] Di Stefano Antonella, Morana Giovanni, Zito Daniele. QoS Management in Grid, Symposium of Consorzio COMETA Palermo, Italy Decembar 2007

A Meta-Heuristic Based Scheduling Algorithm for Grid Environments

Marilena Bandieramonte[1,2], Antonella Di Stefano[2] and Giovanni Morana[2]
[1]*Dept. of Computer Science and Telecommunication Engineering, Catania, Italy*
[2]*INFN, Catania, Italy*
{*mbandiera, ad, gmorana*}*@diit.unict.it*

***Abstract*—The Grid middlewares are today among the most popular and applied supports for e-science applications. The main challenge in these systems is to share heterogeneous resources made available by different organization through different platforms. Grids offer different types of resources belonging to scalable groups of research called Virtual Organization. Job scheduling is one of the most crucial issues in a Grid environment because it strongly affects the performance of the whole system. In literature there are several algorithms that try to obtain the best performance for some specified requirements. We propose the Alienated Ant Algorithm, a meta-heuristic based scheduling algorithm inspired to ACO strategy, to faces the scheduling problems in a Grid environment. Our solution takes into account the behaviour of an ideal alienated ant that stays away from tussle and prefers to search the food following the paths where it can found the minimum number of other ants. The goodness of the proposed algorithm, in terms of load balancing and average queue waiting time, has been evaluated by mean of a wide campaign of simulations carried out on the real scenario of the PI2S2 Grid infrastructure.**

***Index Terms*—scheduling algorithms, Grid computing, simulation, ant colony optimization, alienated ant**

I. Introduction

Grid systems are distributed, high-throughput and very dynamic environments. The aim of these systems is to aggregate and share heterogeneous resources which belong to different administrative domains through different platforms; actually the Grids are among the most widespread and applied supports for e-science applications and they are also becoming an important support for e-business. There are many open challenges in Grid computing; some of these are related to the sharing, the integration, the virtualisation and the management of the distributed resources. In fact, the target of a Grid is not only to put together different resources, but is to aggregate and share them, also providing the aggregation of the capacity of the single resource, offering a system with a higher capacity, that is able to run applications faster. Therefore, resources management with a good scheduling algorithm is a fundamental part in the development of Grid middlewares; job scheduling is one of the most crucial issues in a Grid environment because it strongly affects the performance of the whole system. In literature there are several algorithms that try to obtain the best performance for some specified requirements for scheduling in parallel architectures; so the scheduling problems in a Grid is substantially different. This is the focus of this paper: try to face the issues characterizing the scheduling problem in Grids. To do that we propose the Alienated Ant Algorithm scheduling policy. This scheduling strategy is based on a meta-heuristic algorithm freely inspired on the ACO theory, that emulates the natural intelligent behaviour of ants in looking for food by following the pheromone traces of the other ants. In our work we examined a real Grid infrastructure, the one realized in the PI2S2 project and shared between the Sicilians Re-

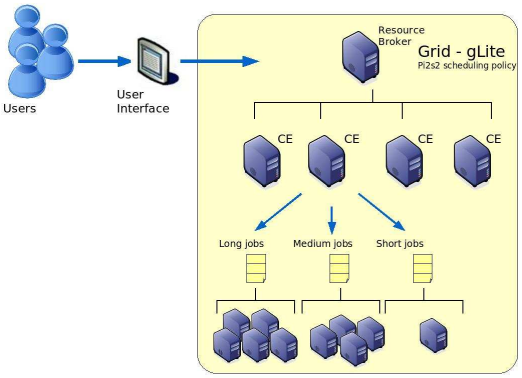


Fig. 1. The PI2S2 scheduling policy.

search Organizations taking part of COMETA, a multi-entity consortium for the adoption and promotion of advanced computing technologies [4]. We have examined the scheduling policy used in this Grid and then we compared the two algorithms evaluating the performance. The Section II analyses the PI2S2 case study, Section III presents thegLite implementation of the Alienated Ant Algorithm, Section IV presents the experimental results of the paper, finally, Section V concludes the work.

II. Problem Statement: the PI2S2 case study

This section presents the case study considered in our research. The scenario that we have taken into consideration is that of the Sicilian Grid belonging to the COMETA consortium and carried out within the project PI2S2. This Grid infrastructure is distributed in Sicily on three main nodes in the following cities: Catania, Messina and Palermo. It consists of three Resource Brokers, each of which is connected to a different number of Computing Elements. Each Computing Element is also the manager of a cluster of Worker Nodes of variable number. The middleware used is gLite v. 3.0.0 that realizes a scheduling policy that takes place on two different levels: a first level of scheduling occurs on the broker, a second level is on Ce.

The Fig. 1 describes the scenario in question: as is evident from the figure, users access the Grid via a User Interface (UI), and so may submit their jobs to the system and to the Broker, who will deal with the job schedule on the appropriate resources, making the entire operation transparent to the user. In the figure is highlighted the presence of the gLite middleware. The job scheduling system of PI2S2 Grid works as follows: at the Resource Broker level is applied a Round Robin policy. Then each Ce mantains three queues: one for long jobs, one for the medium and one for small jobs. Each queue is associated with different Wns. The scheduling policy then reserves some Wns only to perform a certain job type. In the PI2S2 the Wn are reserved by the relevant Ce in the following percentage: 50% for long jobs, 40% for medium jobs and the 10% for short jobs. This scheduling strategy, however, presents, among others, a limit: it does not take into account the dynamic nature of the Grid. The Grid environment, as asserted previously, is an heterogeneous and very dynamic one. For this reason, our work suggests, like alternative, a meta-heuristic, adaptive-like, algorithm for the scheduling, which takes into account the changes in the Grid, and that is able to adjust its behaviour basing on these. Our scheduling policy, like it can see in Fig. 2, moves the management of the job queues: the Resource Broker maintains a single queue which keeps track of informations related to the individual Ce and the scheduling choices made. Each Ce, in turn, maintains a queue with a single entry for each Wn. These queues keep track of scheduling choices made and are used by the scheduling policy to decide the mapping of job to resources, in order to maximize the load balancing and the throughput of the system at both levels of scheduling.

III. The gLite implementation of the Alienated Ant Algorithm

As said previously, the AAA is a scheduling algorithm originated from some considerations made on the Ant Colony Optimization strategy.

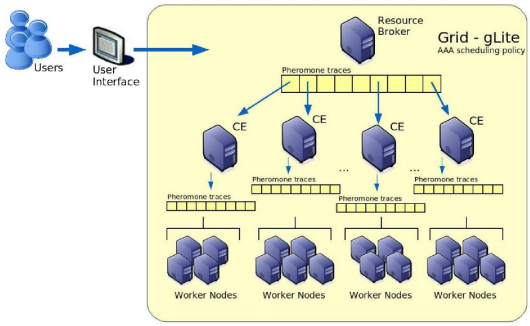


Fig. 2. The AAA scheduling policy.

This is a bio-inspired technique grows out of the observation of the natural behaviour of a colony of ants; proposed in 1992 by M. Dorigo et al [1], [2], [7], [8], it is used successfully to solve complex combinatorial and multi-constraint optimization problems. The strong point of ant colony's behaviour is the *indirect communication* through *pheromone* trails; the pheromone is a chemical substance that is able to cause specific reactions in other individuals of the same species that come into contact with it. Thanks to pheromone, each ant is able to coordinate its decisions and movements with the other ants in order to follow out some works related to colony survival.

The Alienated Ant Algorithm is born as an answer to a question that we set during the analysis of the problem: can you use the ACO strategies to implement the scheduling algorithms on the Grid in order to optimize the performance? Since the single users of the Grid and system administrators have different requirements, sometimes conflicting with each other, there is no single target for the scheduling; from the user point of view, it is important that the system have a short response time, while the administrator of the Grid, on the other hand, is more interested in optimizing the use of computational resources available in the system. For these reasons, among the targets of a good scheduling algorithm, falls surely that of achieve optimal distribution of load to ensure good levels of load balancing and adequate response times.

Taking into consideration the arguments made above, we asked if we could take advantage of the behavior of an hypothetical "alienated" ant for which the pheromone does not carry an attractive force, but a repulsive one. This ant, just because alienated, have an unusual behaviour: it evaluates the pheromone trails in inverse manner. A tenuous pheromone trail on a path means that few ants populate the path or that is a long time since the last ant has chosen it; this consideration makes the path more attractive than those with a stronger trace.

Thus, the traditional ACO algorithm, which tends to convey all the ants on the same path, is transformed into an algorithm (the AAA) that distributes the jobs on every paths, taking into account all the solutions space. The AAA, if applied as scheduling policy in a Grid environment, allows to identify the least loaded computational resource among ones available in Grids. This scheduling algorithm, referring to the steps describing the functioning of generic ACO, performs the following steps:

- *Solution construction step*
- *Solution selection step*
- *Pheromone trail update step*
- *Pheromone trails evaporation step*

Running these phases iteratively, a near-optimal solution for a given problem can be found. For more details see [14].

The middleware used in the scenario on wich we have focused our attention in this work, is gLite v. 3.0.0.1. This middleware splits the management of scheduling on two levels: RB level and CE level. These two levels also have different purposes: the first level tries to distribute as fairly possible job on the resources, the second level, however, tries wherever possible to maximize the throughput of the system. Apply the AAA to the gLite middleware is therefore split decisions on scheduling of two different levels in order to build two-step solution to a problem. In our work we decided to implement the AAA at both levels of scheduling, so as to

ensure the dynamism of the scheduling policy, which is capable of adapting to changes in the system. So the algorithm constructs the solution in two steps: the first is on the path from RB the CE, the second step is on the path from CE to WN.

IV. Methodology and Experimental Results

In this section we propose the experimental results of the simulations carried out to test the behaviour of our algorithm. The performance evaluations and the comparisons with other solutions have been made using the SimGrid [15] simulator, a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments.

As said, the simulated scenario reproduces the real Grid infrastructure of the COMETA's PI2S2 project. The Grid infrastructure and the jobs flow model have been modelled considering the information related to (i) the types and the distribution of resources in the real PI2S2 Grid infrastructure and to (ii) the dimension of waiting queues of different types of jobs typically submitted by the users, actually observed. To evaluate the performance of the algorithms, in this work, it was decided to assess the Average Queue Waiting Time and the Load Balancing. These metrics take in consideration two different point of view: user and administrator perspectives. In the simulation, as well as to compare the two algorithms described above (*AAA* and *PI2S2*), was taken into account a third scheduling algorithm: the random scheduling. This algorithm (in the following *Random*) is used only as a benchmark to show that the adoption of a scheduling policy is a necessary condition to obtain good performance.

A. Experimental results

In this subsection are presented the results obtained from the simulations, by subjecting the system to different conditions. The system was subjected (undergone) to two different loading

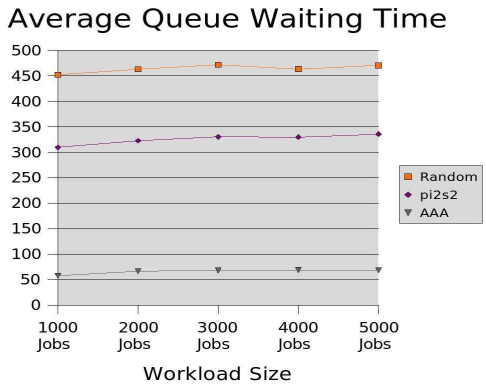


Fig. 3. $aqwt$ with the change in workload size.

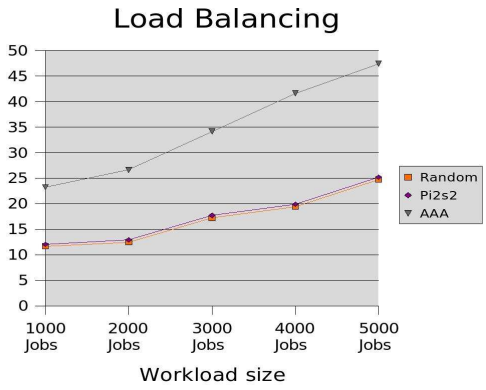


Fig. 4. σlb with the change in workload size.

conditions: in the first simulations campaign, we varied the workload size, in the second its composition.

In the first simulation we tested the behaviour of the algorithms by subjecting the system to increasing workloads: from 1000 to 5000 jobs, maintainig constant the ratio (15-75-10) among different jobs type componing the workload. The Fig. 3 and the Fig. 4 show the results of the simulations. In the first figure is showed the Average Queue Waiting Time for the three algorithms: the trends of the three curves are almost constant, but they run on completely different levels. The *AAA* has the best outcome by offering waiting times with very low values compared to *PI2S2* having values 6 times larger.

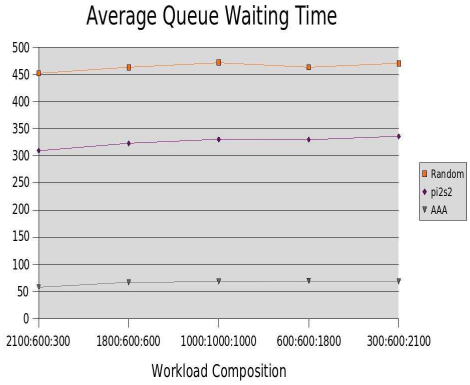


Fig. 5. $aqwt$ with the change in workload composition.

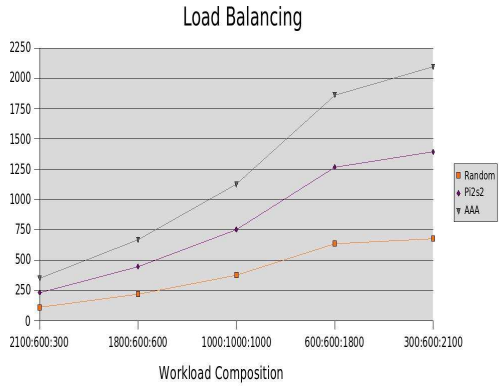


Fig. 6. σlb with the change in workload composition.

The second figure shows, however, the standard deviation of the load balancing. Here too the AAA has the best performance, improving its behavior with increasing of the workload than the other algorithms, thus succeeding to support high workloads. In the second simulation we have tested the bahaviour of the algorithms by subjecting the system to different workload composition: the proportion beetween short, medium and long job it was varied as it shows in the Fig. 5 and in the Fig. 6. Generally, the results described confirm what has been observed previously: also in this case, the AAA presents the best behaviour in terms of aqwt and in terms of load balancing.

V. Conclusion

In this paper the scheduling issue in a Grid was faced. For the purpose, it was proposed a meta-heuristic algorithm: the Alienated Ant Algorithm. The main characteristic of the AAA is an inverse evaluation of pheromone trails that allows to identify the least loaded computational resource among ones available in Grids. This algorithm was applied to the real Pi2S2's Grid and it was compared with the scheduling policy used in this environment. As shown by the results of simulations, our solution improves the performance reducing the jobs average queue waiting time and maintaining a good load balancing capability, in respect with the PI2S2 scheduling algorithm.

References

[1] Marco Dorigo and Christian Blum, Ant colony optimization theory: a survey, Journal Theoretical Computer Science, Volume 344 , Issue 2-3. On pages: 243 - 278, 2005.

[2] Kwang Mong Sim Weng Hong Sun, Ant colony optimization for routing and load-balancing: survey and new directions,IEEE Trans. on Systems, Man and Cybernetics, Part A, Volume: 33, Issue:5. On page(s): 560-572, 2003.

[3] Thomas Stutzle and Holger H. Hoos: MAX-MIN Ant system, Future Generation Computer Systems v.16 n.9. On page(s): 889-914, 2000.

[4] Consorzio Cometa http://www.consorzio-cometa.it

[5] M. Dorigo, V. Maniezzo, A. Colorni: Ant system: optimization by a colony of cooperating agents, IEEE Trans. on Systems, Man and Cybernetics, Part B, Volume: 26, Issue: 1. On page(s): 29-41, 1996.

[6] M. Dorigo, L.M.Gambardella: Ant colony system: a cooperative learning approach to the travelling salesman problem, IEEE Trans. on Evolutionary Computation, Volume: 1, Issue: 1. On page(s): 53-66, 1997.

[7] D. Merkle, M. Middendorf, H. Schmeck: Ant colony optimization for resource-constrained project scheduling, IEEE Trans. on Evolutionary Computation, Volume: 6, Issue: 4, 2003.

[8] C. Blum, M. Sampels: An ant colony optimization algorithm for shop scheduling problems, Journal of Mathematical Modelling and Algorithms Volume 3, Number 3 Springer Netherlands, 2004.

[9] Thomas Stützle and Marco Dorigo: ACO algorithms for the quadratic assignment problem, New ideas in optimization, isbn:0-07-709506-5. On page(s): 33-50, McGraw-Hill Ltd., UK, 1999.

[10] Carl Kesselman and Ian Foster and Steven Tuecke: The Anatomy of the Grid - Enabling Scalable Virtual Organizations, International Journal of High Performance Computing Applications Volume 15, Issue 3. On page(s): 200 - 222, 2001.

[11] I. Foster and C. Kesselman and J. Nick and S. Tuecke: The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Open Grid Service Infrastructure WG, Global Grid Forum, 2002.

[12] I. Foster and C. Kesselman: The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishers(ISBN: 1-558660-475-8), 1999.

[13] Daniel A. Menasc and Emiliano Casalicchio: QoS in Grid Computing IEEE Internet Computing, 8(4). On page(s): 85-87, 2004.

[14] M. Bandieramonte, A. Di Stefano and G. Morana: An ACO Inspired Strategy to Improve Jobs Scheduling in a Grid Environment. ICA3PP. On page(s) 30-41, 2008.

[15] SimGrid project http://simGrid.gforge.inria.fr/

[16] Fangpeng Dong and Selim G. Akl: Scheduling Algorithms for Grid Computing: State of the Art and Open Problems, Technical Report No. 2006-504, 2006.

[17] Yi JIAN and Yanbing Liu: The State of the Art in Grid Scheduling Systems, Third International Conference on Natural Computation, 2007.

[18] Sergey Zhuk and Andrey Chernykh and Arutyun Avetisyan and Sergey Gaissaryan and Dmitry Grushin and Nikolai Kuzjurin and Alexey Pospelov and Alexander Shokurov: Comparison of Scheduling Heuristics for Grid Resource Broker, Proceedings of the Fifth Mexican International Conference in Computer Science, 2004.

[19] Noriyuki Fujimoto and Kenichi Hagihara: A Comparison among Grid Scheduling Algorithms for Independent Coarse-Grained Tasks, Proceedings of the International Symposium on Applications and th Internet Workshops(SAINTW'04), 2004.

[20] Jiayi Zhou and Kun-Ming Yu and Chih-Hsun Chou and Li-An Yang and Zhi-Jie Luo: A Dynamic Resource Broker and Fuzzy Logic Based Scheduling Algorithm in Grid Environment, B. Beliczynski et al. (Eds.): ICANNGA 2007, Part I, LNCS 4431. On page(s): 604-613, 2007.

[21] T.L. Casavant and J.G. Kuhl: A taxonomy of scheduling in general-purpose distributed computing systems, IEEE Transactions on Software Engineering, February 1988 (Vol. 14, No. 2). On page(s): 141-154, 1988.

[22] F. Berman, R. Wolski, H. Casanova, W. Cirne, H. Dail, M. Faerman, S. Figueira, J. Hayes, G. Obertelli, J. Schopf, G. Shao, S. Smallen, N. Spring, A. Su and D. Zagorodnov: Adaptive Computing on the Grid Using AppLeS, IEEE Trans. on Parallel and Distributed Systems (TPDS), Vol.14, No.4. On page(s): 369 - 382, 2003.

[23] Hongzhang Shan, Warren Smith: Job Scheduling in a Heterogeneous Grid Environment, Computational Research Division. Computer Sciences Corporation

[24] Pavel Fibich, Ludek Matyska and Hana Rudova: Model of Grid Scheduling Problem, 2005, American Association for Artificial Intelligence (www.aaai.org).

[25] Alain Andrieux, Dave Berry, Jon Garibaldi, Stephen Jarvis, Jon MacLaren, Djamila Ouelhadj and Dave Snelling: Open Issues in Grid Scheduling, UK e-Science Technical Report Series ISSN 1751-5971

[26] Global Grid Forum http://www.Gridforum.org

[27] Globus toolKit http://www.globus.org

[28] gLite middleware http://www.egee.org

[29] Omii-UK middleware http://www.omii.org

[30] Volker Hamscher, Uwe Schwiegelshohn, Achim Streit and Ramin Yahyapour: Evaluation of Job-Scheduling Strategies for Grid Computing, Lecture Notes in Computer Science, ISSN 0302-9743 (Print) 1611-3349

[31] Y. Zhu: A survey on Grid scheduling systems, Department of Computer Science, Hong Kong University of science and Technology, 2003.

[32] Hongzhang Shan, Leonid Oliker, Warren Smith and Rupak Biswas: Scheduling in Heterogeneous Grid Environments: The Effects of Data Migration, in Proc. of ADCOM2004: International Conference on Advanced Computing and Communication, Ahmedabad Gujarat, India, 2004.

[33] Hongzhang Shan, Leonid Oliker, Warren Smith and Rupak Biswas: High-Performance Schedulers, chapter in The Grid: Blueprint for a Future Computing Infrastructure, Morgan Kaufmann Publishers, 1998.

[34] Kwang Mong Sim and Weng Hong Sun: Ant Colony Optimization for Routing and Load-Balancing: Survey and New Directions, Systems, Man and Cybernetics, Part A, IEEE Transactions - Volume: 33. On page(s): 560- 572, 2003.

[35] Ruud Schoonderwoerd, Owen Holland and Janet Bruten: Ant-like Agents for Load Balancing in Telecommunications Networks, Proceedings of the first international conference on Autonomous agents, Marina del Rey, California, United States, 1997.

[36] Chiang, C.-W. Lee, Y.-C. Lee, C.-N. Chou, T.-Y: Ant colony optimization for task matching and scheduling, Computers and Digital Techniques, IEE Proceedings - Volume 153. On page(s): 373-380, 2006.

[37] Daniel Merkle, Martin Middendorf, Hartmut Schmeck: Ant Colony Optimization for Resource-Constrained Project Scheduling, Proceedings of the Genetic and Evolutionary Computation Conference, 2000.

[38] Vittorio Maniezzo, Matteo Roffilli: Very Strongly Constrained Problems: an Ant Colony Optimization Approach, Cybernetics and Systems, Volume 39, Issue 4 May 2008. On page(s): 395 - 424

[39] Graham Ritchie and John Levine: A Hybrid Ant Algorithm for Scheduling Independent Jobs in Heterogeneous Computing Environments, in: Third Workshop of the UK Planning and Scheduling Special Interest Group (PLANSIG2004), 2004.

[40] Stefka Fidanova and Mariya Durchova: Ant Algorithm for Grid Scheduling Problem, LNCS 3743. On page(s): 405-412, 2006.

[41] Hui Yan, Xue-Qin Shen, Xing Li and Ming-Hui Wu: An Improved Ant Algorithm for Job Scheduling in Grid Computing, Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 2005.

[42] C. Blum and A. Roli: Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison, ACM Computing Surveys, Vol. 35, No. 3. On page(s): 268-308, 2003.

[43] Pavani, G.S. Waldman, H.: Grid Resource Management by means of Ant Colony Optimization, Broad-

band Communications, Networks and Systems, 2006. BROADNETS 2006. On page(s): 1-9, 2006.
[44] Artur Andrzejak, Artur Andrzejak, Sven Graupner, Vadim Kotov, Holger Trinks: Algorithms for self-organization and adaptive service placement in dynamic distributed systems, HPL-2002-259 20020923
[45] M. Dorigo, L.M. Gambardella: Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evol. Comput. 1. On page(s): 53-66, 1997.
[46] L.M. Gambardella, M. Dorigo: Ant colony system hybridized with a new local search for the sequential ordering problem , INFORMS J. Comput. 12. On page(s): 237-255, 2000.
[47] M. Reimann, K. Doerner, R.F. Hartl: D-ants: savings based ants divide and conquer the vehicle routing problems, Comput. Oper. Res. 31. On page(s): 563-591, 2004.
[48] den Besten ML, Stutzle T, Dorigo M.: Design of iterated local search algorithms: An example application to the single machine total weighted tardiness problem, In Proceedings of EvoStim01, Lecture Notes in Computer Science, Berlin. On page(s): 441-452, Springer, 2001.
[49] Thomas Stutzle: An ant approach to the flow shop problem, In Proceedings of the 6th European Congress on Intelligent Techniques & Soft Computing, 1998.
[50] Christian Blum, Michael Sampels: Ant colony optimization for fop shop scheduling: A case study on different pheromone representations, In Proceedings of the 2002 Congress on Evolutionary Computing. On page(s): 1558-1563, 2002.
[51] Gao, Y., Rong, H., Huang, J.Z.: Adaptive Grid job 1215 scheduling with genetic algorithms. Future Generation. Computer. Syst. 21, 151-161, 2005.
[52] Cao, J., Spooner, D.P., Jarvis, S.A., Nudd, G.R.: Grid 1218 load balancing using intelligent agents. Future Generation. Computer. Syst. 21, 135-149, 2005.
[53] Spooner, D.P., Jarvis, S.A., Cao, J., Saini, S., Nudd, 1221 G.R.: Local Grid scheduling techniques using performance prediction. IEE Proc. E 150, 87-96, 2003.
[54] Yang, L., Schopf, J.M., Foster, I.: Conservative Sched-1224 uling: Using Predicted Variance to Improve Scheduling Decisions in Dynamic Environments. In: Proceedings of the 2003 ACM/IEEE conference on Supercomputing, pp. 31-47, 2003.

M. Bandieramonte took the Laurea degree in Computer Science Engineering from the University of Catania (Italy) in 2007. Her research activity focuses on metaheuristics and scheduling algorithm in Grid Computing.

A. Di Stefano received the Laurea degree in Electrical Engineering from the University of Catania (Italy) in 1983. Her research interests have been mainly in the area of parallel and distributed systems, including applicative scenarios, models, software design, architectures and protocols she research project in the field of distributed systems. Currently, she is a full professor of operating systems in the Engineering Faculty of the University of Catania. She is currently working on multiagent systems, Grid and P2P, with special emphasis given to distributed scheduling, QoS and WS.

G. Morana has received his Ph.D. degree (2009) in Computer Science and Telecommunication Engineering at University of Catania, where he took his Laurea degree in Computer Science Engineering. His research interests are mainly in the area of parallel and distributed systems, including applicative scenarios, models, architectures and communication protocols, with particular references to Grid Computing and peer-to-peer systems.

Dissecting BitTorrent using Grid

C. Pistagna[1], S. Riccobene[1]
[1]*Dipartimento di Matematica e Informatica, Università di Catania*
Viale A.Doria, 6 I-95125 Catania, Italy
{*pistagna,sriccobene*}*@dmi.unict.it*

***Abstract*—Data center applications are still a challenging issue for large scale distributed computing systems. Literature offers many proposal striving for the subject of fast and efficient data delivery. BitTorrent is a wide-used file distribution protocol, that goes over the lack of other peer-to-peer systems, achieving the expected goal through local network and internet. The base idea of BitTorrent is a strong cooperation between downloaders, using an incentive mechanism based on the tit-for-tat strategy. Our work focuses on a "two stages" evaluation of the protocol, using a special crafted grid framework. In the first stage, we manipulate the torrent size, the number of peers involved, their bandwidth and other distinctive protocol parameters. The resulting BitTorrent session is submitted to the grid, that uses computing worker nodes as torrent peers and trackers. In the second stage, the above starting parameters are used as input for a BitTorrent discrete event simulator, especially designed for this purpose. Results are compared each other, giving us a fine grain about the rightness of the discrete event simulator. Moreover, since the simulations are built on top of the grid, we can accurately evaluate the use of BitTorrent as a grid file transfer protocol.**

***Index Terms*—**
COMETA, BitTorrent, NS2, grid.

I. Introduction

BITTORRENT is a popular file distribution system which aims to avoid the lack of other peer-to-peer protocols. The base idea of BitTorrent is cooperation between downloaders, using a strong incentive mechanism based on the tit-for-tat strategy. First of all, a peer gets information about the resource it wants to download using a special metadata file, called "torrent file". The way in which this file is obtained can be safely ignored, since in many cases it is a traditional web download. The torrent file contains the SHA1 signatures for all the chunks composing the desired resource and the location of a central agent, called tracker, which helps the peers to discover each others. The tracker does not organize transfers of the chunks between peers. Instead, they are made locally by peers, according to their information regarding neighbors. BitTorrent does not provide central resource allocation. Each peer is responsible to maximize its own download rate. Peers ask to their neighbors for the transfer of the file chunks, selecting them from a list of peers received by the tracker. The rarest chunks are preferred among all availables ones. Periodically, every peer computes the ratio between upload and download with other peers. If a value falls down under a prefixed threshold, the examined peer will be disconnected (chocked), according to a variant of the tit-for-tat strategy. Discovering of new peers with better performances is achieved through a so called optimistic unchoke algorithm. In this latter case a random peer is unchocked, it does not matter about its upload/download rate.

Switching time between the optimistic unchoked peers is done regularly during the runtime period, giving them enough time to get unchocked by the remote peer in turn. The strategy is changed when the download is completed. In this phase, unchocking is based exclusively on the upload rate to the connected remote peers. Although the choking algorithm is not technically part of the BitTorrent protocol, it

is necessary for optimal performance, since neighbors list is randomly choosen by tracker. Good choking algorithms should guarantee at least two features: resilience to peers that are downloading without contribute (leechers) and fairness in download rates among peers [1]. Every peer has a personal interest rate in sharing its resources. If this value is uniformly distributed among other peers, each one will obtain the maximum personal interest, described in our model as maximized bandwidth and a better compromise between uploaded and downloaded chunks (shared resources). Our goal is to demonstrate that our simulator results are closer to a real BitTorrent session.

II. Related Works

Among the most popular p2p file sharing protocols, BitTorrent has been one of the most used in the last few years. There are many interesting works related to system design and traffic measurements of BitTorrent. In [24], a fluid model is used to study the performance of a real BitTorrent system. In [15], [18], [10] tracker logs are used, in order to reflects and describe the BitTorrent behaviour. In [12], a branching process is used to study the service capacity of BitTorrent in the transient regime. In [3], the authors propose a genetic algorithm based selection strategy to enhance tracker capabilities. [4] decomposes the BitTorrent protocol and evaluate the impact of its core mechanisms, focusing on several important metrics like peer link utilization, file download time and fairness. The model used is a simplified one without network propagation delay and no packet-level dynamics of TCP connections, due to computational complexity of the model. In [19], a graph based multi-torrent model is used to study inter-torrent collaboration. [20] uses a modified BitTorrent client to sniff real torrent swarms characteristics. In [21], the authors compare a centrally scheduled file distribution protocol with a BitTorrent session, comparing and analyzing the performances. In [6], a new trading scheme based on pricing is adopted. The authors show how the stable point of the algorithm provides a fair resource allocation for all the swarm. [23] presents an extensible framework for simulating P2P networks by using message level simulation. A flow model is employed to accomplish accurate estimate of the message behaviour. In [5], the authors analyze protocols for disseminating a collection of data blocks over multiple network topologies, without modeling variation of bandwidth or BitTorrent tit-for-tat strategy.

In our opinion, this is the first attempt to study BitTorrent traffic measurements and protocol behaviour by close comparing real torrent statistics with the ones provided by a grid based simulator. Our contribute focuses on the "case studies" and the way in which they are generated. Moreover, we propose new protocol suggestions in order to improve the overall torrent life. In a first step, we manipulate the torrent size, the number of peers involved and their bandwidth as well as other distinctive features, using a special crafted framework. The resulting BitTorrent session is submitted to the grid, that uses computing worker nodes as torrent peers. In a second time, the same crafting parameters are used to run a simulated swarm under the discrete event simulator NS2. Results comparison give us a very fine grain about the rightness of the discrete event simulation, letting us to tune and investigate deeper in the BitTorrent protocol and its optmizations.

The rest of this paper is organized as follow: in section III, we introduce the BitTorrent grid framework, its design and the way in which the torrent jobs are submitted to the grid. In section IV, we introduce our discrete event simulator, the way in which it simulates a BitTorrent swarm and its startup parameters. We illustrate the results of our work and in section V we discuss about considerations and related future works.

III. The BitTorrent grid framework

To evaluate how good and reliable is our NS2 simulator, we developed a BitTorrent swarm environment based on the open source client CTorrent. The environment uses grid computing

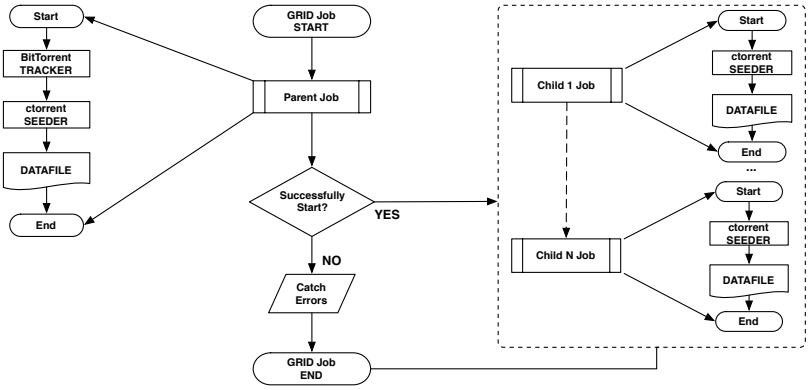


Fig. 1. Flowchart diagram, illustrating the flow of the Grid Framework. First, the parent job (seeder) is started. Downloaders clients are created as child processes, using the DAG-Jobs Grid abstraction.

facilities to submit, log and monitor torrent jobs. CTorrent is a BitTorrent client implemented in C++: its lightweight and modular design results in a fast and cross-platform product, suitable to run on any Unix-like system including FreeBSD, Linux, Mac OS X, Solaris.

Thanks to its runtime flexibility, we can start quickly a new torrent session with upload, download and ratio parameter variables. All these features make CTorrent the perfect candidate to build a real-life BitTorrent simulation framework. The way in which the framework works is straightforward: first, a parent job is submitted. It is responsable to create a dummy datafile with specified dimensions. Once the file is created, the parent job starts a full-featured Apache server. Finally, it creates and submits the new torrent to the tracker, using the CTorrent client as seeder and the above generated datafile as datasource. Customized runtime parameters, are employed to start the new BitTorrent session: *downloadBand*, *uploadBand* and *seedTime*. They are obtained with a pareto random generator, configured with lower bound and upper bound values. If the parent job is successfully loaded, then the child jobs are started, according to their runtime parameters. Fig.1, shows a diagrams representation of our BitTorrent framework. Child jobs inherit from parent the same input file, the torrent metadata. The behaviours of child processes, in terms of download and upload bandwidth, ratio, join time and leave time, are also controlled by a random generator, appropriately filled with startup values for each job instance.

The overall job behaviour, including parent and child jobs, is monitored and controlled using the DAG-Jobs GRID extension available with the gLite GRID framework. A DAG (Directed Acyclic Graphs) represents a set of jobs in which input, output and execution of one or more of them, depend on one or more jobs. The Edges (arcs) represent the dependencies and the jobs are nodes (vertices) in the graph. There can be a set of common attributes valid for all jobs: in our framework, the input sandbox containing the torrent metadata file generated above, is shared among jobs. Further, the dependencies, which schedule the order of execution, are defined as pairs of type:

$$d = \{A, B\} \tag{1}$$

The (1) states that B cannot be executed before A has finished. Each element could also be a pair itself. The overall logs about both tracker and clients are saved in an output file, returned as a standard text. The joining and leaving behaviour of nodes is managed by a logistic

function, as follow:

$$P(t) = \frac{1}{1 + e^{-t}} \qquad (2)$$

The (2) models the S-curve of growth of some set *P*, where *P* represents a population and *t* is the time. The initial stage of growth is approximately exponential. Then, as saturation begins, the growth slows and at maturity, growth stops. As noticed, there is a closer analogy with a typical swarm life, in which the number of interested nodes grow slowly in the first time, increase exponentially in the middle part and reach maturity at the end, as shown in fig. 2.

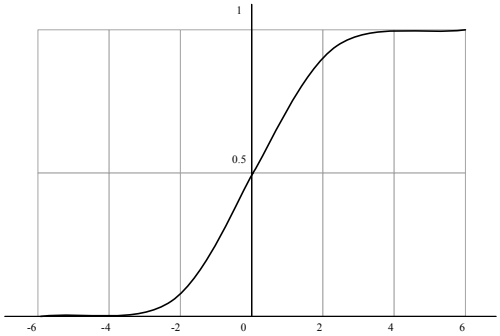


Fig. 2. Standard logistic sigmoid function. The initial stage of growth is approximately exponential; then, as saturation begins, the growth slows, and at maturity, growth stops.

In a first time, we conducted a set of experiments using runtime parameters to simulate a collaborative swarm without selfish nodes. All participants have asymmetric upload and download bandwidth, simulating a standard ADSL line with speed ranging from 512Kbit/s to 2Mbit/s. Fig. 3, shows a comparison of results obtained by launching simulations with only collaborative nodes and both selfish and collaborative ones. All simulations employ 40 clients and 1 seeder. Selfish nodes have no upload bandwidth and they leave the swarm just after the download is completed. In a selfish population, the swarm evolution, in terms of peers becoming seeders, is slower than the one with all collaborative nodes.

The overall bandwidth usage shows significant changes as we use only collaborative or

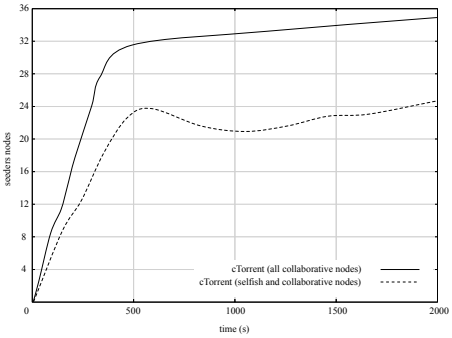


Fig. 3. Number of seeders in a cTorrent session. After the initial stage of growth, as saturation begins, the growth slows, and at maturity, growth stops.

both selfish and collaborative nodes. Fig.4 depicts bandwidth usage for the simulated scenarios.

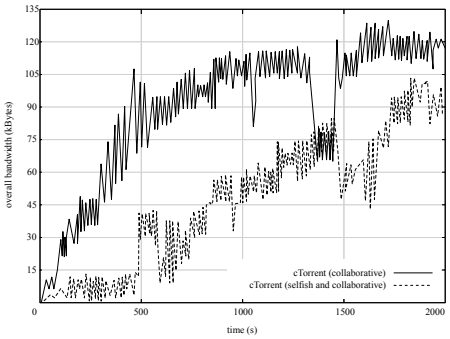


Fig. 4. Bandwidth usage in a cTorrent session. Collaborative swarm is more efficient in term of bandwidth usage than the one with selfish nodes.

In the next section, we compare these results with the ones provided by NS2.

IV. The BitTorrent NS2 Simulator

BitTorrent is composed of several mechanisms that interact in many complex ways, depending on the local context of every peer. The second stage of this study exploits a discrete event simulator written in NS2 [2]: it provides the flexibility of carefully controlling behaviour mechanisms and environmental conditions above all the interacting nodes as well

as many of the associated BitTorrent mechanisms (local rarest first, tit-for-tat, etc.). This approach allows us to explore system performance in complex and different scenarios. Since our simulations are cpu and memory expensive, we conducted them using the COMETA Grid infrastructure. To compare the real swarm results obtained in III, a BitTorrent simulator was developed from scratch using the objective-TCL language [14], that is the de-facto standard programming language for NS2.

We modeled a generic peer by extending Application/TcpApp class. Send and Receive methods reflect the real behaviour of packets into the network and pass them to the application level. Send method pushes data into events stack, while the receive one gets and parses arrived messages, acting as described into the BitTorrent protocol. Moreover, this class is responsible to implement the tit-for-tat strategy (fig. 5).

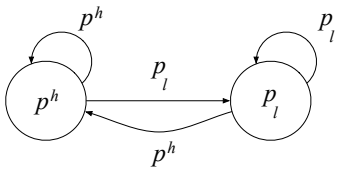


Fig. 5. Tit-for-tat automaton.

An agent using this strategy will initially cooperate, later its behaviour will reflect the opponent's previous actions. If the opponent previously was cooperative, the agent is cooperative; if it was selfish, the agent is selfish. This is similar to reciprocal altruism in biology. The tit-for-tat variant used in BitTorrent is called optimistic unchoking. Each peer has a limited number of upload slots to allocate for other peers. Cooperation is achieved when upload bandwidth is exchanged for download bandwidth. Consequently, when a peer saturates its upload bandwidth, it will use a tit-for-tat strategy. Optimistic unchoking corresponds very strongly to always cooperating on the first move in prisoners dilemma [8]. Periodically, a peer will allocate an upload slot to a randomly chosen peer (unchoke). This allows to search for more cooperative peers and gives a second chance to previously non-cooperative ones.

Node type, bandwidth, join time, leave time and other parameters are independent characteristics, selected at runtime with the $buildCharacter$ method. In order to obtain realistic arrival patterns and building parameters, we used a Poisson distribution function. Moreover, we modeled class variables representing the local-rarest-first block abstraction as well as upload/download status. Using this approach, we can analyze many important characteristics that other studies ([4], [15], [18], [10]) do not consider. Network propagation delay and the dynamics of TCP connections are two crucial features that our simulator takes into account, thanks to the built-in features of NS2. Furthermore, other studies assume that the bottleneck link is either the uplink of the sending node or the downlink of the receiving node; in contrast, our work distinguishes between individual peer bottlenecks, increasing the detail of the simulation. We evaluate the efficiency of BitTorrent in terms of link usage, download time and fairness. In a first time, several simulations was launched, in order to understand how near to a real BitTorrent swarm was our simulated one. Later, we inject in the swarm some malicious peers that are programmed to behave as leechers (ie. nodes that download without uploading). Obtained results are compared with the real one discussed in III. The number of seeders is a health status index of our swarm: more seeders indicate that our swarm will grow up rapidly and since seeders are not driven by tit-for-tat policy, they give priority exclusively to fastest and incomplete peers.

Results in fig. 6 depict the behaviour of our simulator with both collaborative and selfish nodes: they show how nodes become seeders in a cooperative swarm and in presence of selfish nodes. We can observe how a population with all collaborative nodes grows faster than one with both selfish and collaborative nodes.

Let n be the number of peers in our swarm. For each peer *i*, let p_i be its percentage of com-

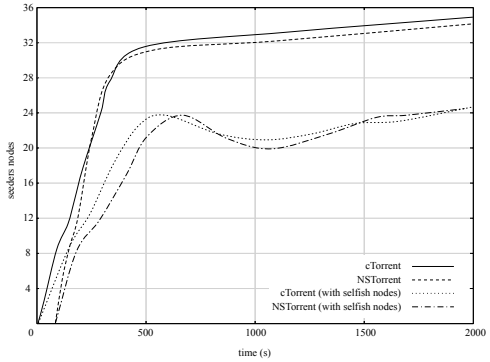


Fig. 6. Number of seeders in a cooperative swarm and in a both selfish and collaborative one.

pletion. We define the value of overall percent completion as follow:

$$OPC = \frac{\sum p_i}{n} \tag{3}$$

OPC is another degree of how good is the health of our swarm: if it rise up fast and maintains its level for the remaining simulation time, then our torrent is in a good health status (fig. 7).

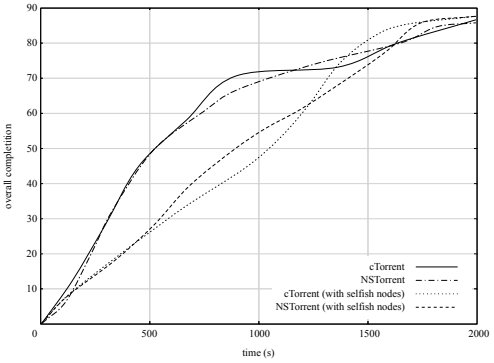


Fig. 7. Overall percent completion in a cooperative swarm and in a both selfish and collaborative one.

A deeper study on simulator outcomes shows how overall swarm health status is strictly connected to its number of peers: larger and collaborative swarms correspond to huge overall bandwidth values (fig. 8 and fig. 10).

Moreover, a strongly selflessness between peers is directly related to the number of nodes in the swarm: peers quickly drop connections to slow or low popular torrents with too many selfish nodes. At a glance, a larger swarm with many collaborative nodes translates into a more fair swarm in terms of bandwidth usage (fig. 10). Thanks to the simulator object model depicted above, we can analyze more in depth the memory status of each peer. Since the memory map of chunks is a matrix with a status flag for each block, we can plot the presence of local rarest pieces as function of time *t*. Local rarest pieces play a key role in the seeding process, giving us another fundamental torrent health index: local rarest pieces decrease faster in a collaborative swarm, making the whole document more available for seeding.

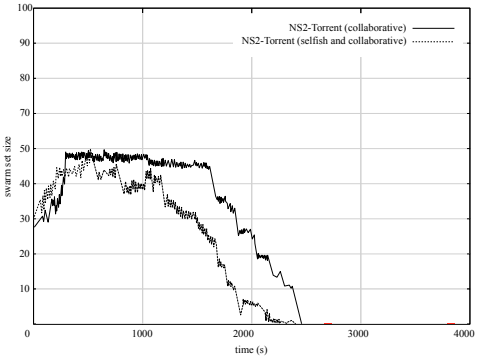
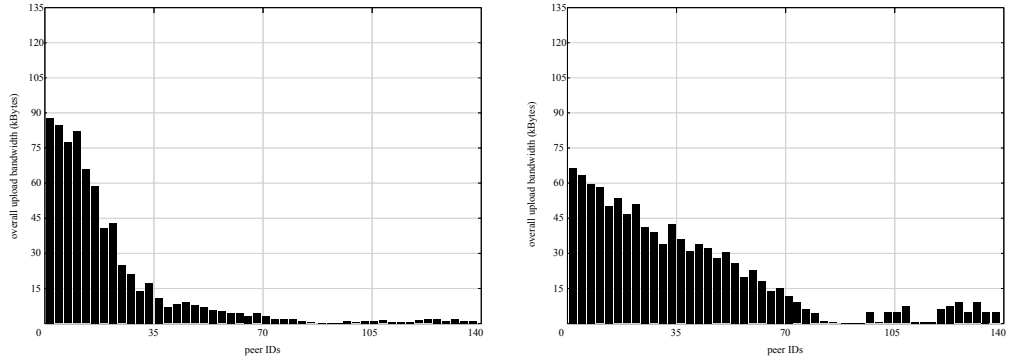


Fig. 8. Swarm size in a BitTorrent session with collaborative nodes and both collaborative and selfish nodes.

V. Conclusions

Our work focused on two main concepts: validating a BitTorrent simulator and building a system framework suitable for deeply dissecting the protocol. As stated above, BitTorrent is one of the most important protocol for large file distribution. However, the presence of selfish nodes can severely compromise the overall efficiency of the protocol, causing slow seeds creation and large rarest pieces presence in the swarm. In a future work, we will investigate deeply into neighbors selections and tit-for-tat variants, in



(a) (b)

Fig. 10. Bandwidth comparison between a mixed selfish/collaborative swarm (a) and a collaborative only (b) swarm.

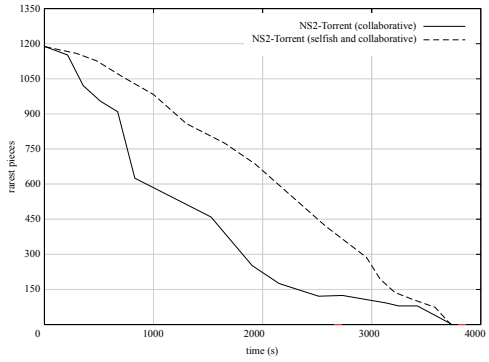


Fig. 9. Rarest pieces in a swarm with all collaborative nodes and both selfish and collaborative nodes.

order to make the protocol more resistant to selfishness variations.

The grid framework discussed in section III gave us an accurate evaluation about using BitTorrent for file distribution over the grid. Therefore, we will investigate more in depth on using the protocol for grid file transfer integration. We believe that, for large data transfer (ie. between storage elements), this protocol is an interesting alternative, among the existing protocols, for bringing down bandwidth wasting.

ACKNOWLEDGMENT

The authors would like to thank E.V. DiStefano for her precious help and support during the last months.

REFERENCES

[1] B. Cohen, *Incentives Build Robustness in BitTorrent*. Workshop on Economics of Peer-to-Peer Systems, 2003.

[2] NS2 Network Simulator, http://www. isi. edu/nsnam/ Workshop on Economics of Peer-to-Peer Systems, 2003.

[3] SGM. Koo, CSG. Lee, K. Kannan, *A genetic-algorithm-based neighbor-selection strategy for hybrid peer-to-peer networks*. Computer Communications and Networks 2004. ICCCN, 2004.

[4] AR. Bharambe, C. Herley, VN. Padmanabhan, *Analyzing and Improving a BitTorrent Networks Performance Mechanisms*. Proceedings of IEEE INFOCOM 2006. IEEE, 2006.

[5] D. Arthur, R. Panigraphy, *Analyzing BitTorrent and Related Peer-to-Peer Networks*. Proceedings of ACM SODA '06. ACM, 2006.

[6] K. Eger, U. Killat, *Bandwidth trading in BitTorrent-like P2P networks for content distribution*. Elsevier, 2008.

[7] W. Yang, N. Abu-Ghazaleh, *GPS: A General Peer-to-Peer Simulator and its Use for Modeling BitTorrent*. IEEE MASCOTS 2005. IEEE, 2005.

[8] A. Rapoport, AM. Chammah, *Prisoner's Dilemma*. The University of Michigan Press, 1965.

[9] D. Erman, D. Ilie, A. Popescu, *BitTorrent session characteristics and models*. HET-NETs, 2005.

[10] N. Andrade, M. Mowbray, A. Lima, G. Wagner, M. Ripeanu, *Influences on cooperation in BitTorrent communities*. ACM SIGCOMM 2005. ACM, 2005.

[11] G. Liu, MZ. Hu, BX. Fang, HL. Zhang, *Explaining BitTorrent traffic self-similarity*. Proc. of the PDCAT2004 - Springer, 2004.

[12] FL. Piccolo, G. Neglia, G. Bianchi, *The Effect of Heterogeneous Link Capacities in BitTorrent-like File Sharing Systems*. PTPSYS2005. IEEE, 2005.

[13] S. Jun, M. Ahamad, *Incentives in BitTorrent induce free riding*. ACM SIGCOMM 2005. ACM, 2005.

[14] P. Bogdanovich, *Objective-Tcl: an object-oriented Tcl environment*. 3rd conference on USENIX Third Annual Tcl/Tk Workshop. ACM, 1995.

[15] M. Izal, G. Urvoy-Keller, EW. Biersack, PA. Felber, A. Al Hamra, L. Garcs-Erice, *Dissecting BitTorrent: Five Months in a Torrent's Lifetime*. Lecture Notes in Computer Science 2004. Springer, 2004.

[16] J. Pouwelse, P. arbacki, D. Epema, H. Sips, *The BitTorrent P2P File-Sharing System: Measurements and Analysis*. Lecture Notes in Computer Science 2005. Springer, 2005.

[17] D. Qiu, W. Sang, *Global stability of Peer-to-Peer file sharing systems*. Computer Communications. Elsevier, 2008.

[18] T. Karagiannis, P. Rodriguez, K. Papagiannaki, *Should internet service providers fear peer-assisted content distribution?*. SIGCOMM 2005. ACM, 2005.

[19] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, X. Zhang, *Measurements, Analysis, and Modeling of BitTorrent-like Systems*. USENIX IMC05. USENIX, 2005.

[20] A. Legout, G. Urvoy-Keller, P. Michiardi *Understanding BitTorrent: An Experimental Perspective*. INRIA France, 2005.

[21] G. Wu, T. Chiueh, *How efficient is BitTorrent?*. Proceedings of SPIE2006. AIP, 2006.

[22] D. Qiu, R. Srikant, *Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks*. SIGCOMM 2004. ACM, 2004.

[23] B. Yang, T. Condie, S. Kamvar, H. Garcia-Molina, *Non-Cooperation in Competitive P2P Networks*. ICDCS 2005. IEEE, 2005.

[24] Y. Yue, C. Lin, Z. Tan, *Analyzing the performance and fairness of BitTorrent-like networks using a general fluid model*. Computer Communications. Elsevier, 2006.

VIDA: A new Virtual Images Deployment Architecture

Paolo Anedda[1], Claudio Guidi[1] and Gianluigi Zanetti[1]
[1]*CRS4, Pula, Italy,* {*paolo.anedda,cguidi,gianluigi.zanetti@crs4.it*}

***Abstract*—Virtualization is an essential technology for building computational infrastructures tailored to the needs of a research organization. Virtual machines enable the specialization of operating systems to particular tasks with hardware resources safely and transparently multiplexed by the vm hypervisor. They also allow the decoupling of the physical computational infrastructure management from the management of the user-visible virtual computing facility. In this paper we describe VIDA, a new software architecture for the deployment of virtual computational infrastructures being currently developed as a part of the Cybersar project. The dynamic construction of specialized virtual clusters is the main mechanism that we use to support large scale applications spanning the resources of multiple sites. The architecture is being built so that it can dynamically allocate resources to applications via a general control plane orchestrating the system computational and network components. The implementation of the architecture is based on the SOA principles. This approach allows us to develop a general architecture for managing virtual clusters in a scalable and flexible manner which is independent from the underlying hardware and system technologies. In particular, we choose to adhere to the W3C style of implementation. The entire system is basically build upon web services and is implemented using standard languages and protocols. We use Xen as the virtualization layer and a web service wrapper to remotely control the management of virtual machines. The logic behind the creation and the management of a virtual cluster is implemented as a set of information workflows of cooperating components, using the Jolie language. Through the use of these workflows the system is able to allocate the physical devices as well as manage virtual resources. VIDA is used within Cybersar to deploy virtual hadoop clusters in a production environment to support bioinformatics experimentation.**

***Index Terms*—VIDA, virtualization, SOA, virtual clusters, xen.**

I. Introduction

RECENT advances in virtualization technologies and the exponential growth in network bandwidth are introducing new dimensions for the computational facilities configuration space and opening interesting and effective new ways to deliver High Performance Computing (HPC) resources to applications.

Virtual machines, such as VMWare, Xen, and KVM, bring to HPC two main gifts. On one hand, they enable the specialization of operating systems to particular tasks with hardware resources safely and transparently multiplexed by the vm hypervisor. On the other hand, virtual machines allow the decoupling of the physical computational infrastructure management from the management of the user visible virtual computing facility, making the former essentially homogeneous and agnostic with respect to the specific applicative context, while putting the latter in the hands of the virtual organization managing the virtual resource. This separation removes many of the difficulties that oppose the effective sharing of computational resource between organizations that are not adhering to very strict common standards for their software stacks.

Virtualization guarantees a high level of flexibility in the dynamic configuration and use of atomic computational resource, which can then be organized and orchestrated to provide virtual clusters that, differently from standard,

general purpose, HPC computing facilities, can be tailored to satisfy the needs of specific virtual organizations.

Virtual clusters can encompass computational resources that are distributed on different, potentially remote sites, current trends on ever increasing available network bandwidth make it possible at least when the structure of the applications run allows to trade latency for pipeline depth to tightly couple geographically distant computational resources in a single computational facility that appears as a coherent entity to its users. Such a constructions, however, requires a much closer coordination between computational and network resources with dynamic network configuration mechanisms, being activate in parallel to the virtual cluster construction.

In this paper we describe VIDA, a new software architecture for the deployment of virtual computational infrastructures being currently developed as a part of the Cybersar project. The dynamic construction of specialized virtual clusters is the main mechanism that we use to support large scale applications spanning the resources of multiple sites. The architecture is being built so that it can dynamically allocate resources to applications via a general control plane orchestrating the system computational and network components. The logic behind the creation and the management of a virtual cluster, that we can represent as a set of information workflows of cooperating components, is demanded to a SOA-based control plane. The entire system is basically build upon web services and is implemented using standard languages and protocols. We use Xen as the virtualization layer and a web service wrapper to remotely control the management of virtual machines.

II. Related works

There are plenty of papers in literature describing the use of Service Oriented Architecture, and particularly the use of web services, for the integration of heterogeneous systems in the world of grid computing and high performance computing.

In [1] the authors describe a Service-Oriented Computing Kit (SOCK), which is an overarching framework covering the key artifacts in planning, modeling, designing, developing, deploying, and managing service-oriented solutions in the enterprise computing space. They analyze all those aspects in details and propose an open source mind-mapping implementation to be used in guiding the migration from the conventional multi-tier web application style to a service-oriented paradigm.

In [2] IBM researchers identify Grid computing and Service Oriented Architecture as the enabling technologies for the increasing request of on-demand operating environments and on-demand virtualization. They clearly illustrate how these technologies can help scientists as well as commercial companies to better satisfy their users' requirements.

[3] provides an in-depth overview of web services, business processes modeling and orchestration languages in the field of grid computing.

Starting from the results presented in [4], in this paper we describe a SOA architecture for the deployment of virtual clusters for high performance computational applications.

Following the work described in [5], we have built a physical resources management system that is capable of machine configuration and reboot, through web service invocation. For the management of virtual resources we used Xen as virtualization layer and web services to control virtual hosts. The same principles are used in [6] but, in contrast with their approach, instead of controlling a pool of virtual nodes organized in so-called "workspaces", we chose to allow the user to directly control each virtual node through the corresponding web service. This allowed us to achieve greater flexibility and better resource allocation.

III. System Architecture

The main functionalities required for deploying virtual clusters are dynamic partitioning of a pool of high performance machines and dynamic creation of specialized virtual resources.

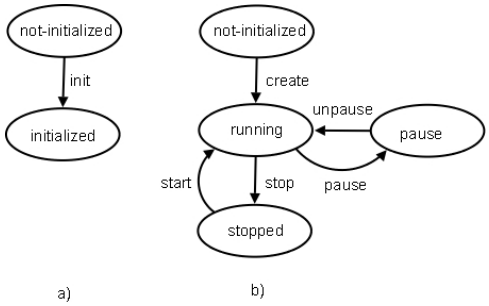


Fig. 1: State diagrams for physical and virtual resources.

In order to achieve this, the software manager must provide the following abstractions:

- It has to allow for the selection of a pool of physical machines where each of them can be initialized in a different way from the other ones. The state diagram reported in figure 1.a, which represents each machine, contains two different states: not-initialized and initialized. A machine can be initialized by performing an *init* action.
- For each physical machine it has to allow for the creation of a pool of different virtual machines which could have different setups. Virtual machines can be started and stopped dynamically. The state diagram which represent them is reported in figure 1.b and it contains four states: not-initialized, stopped, running and pause. State change is triggered by the *create*, *start*, *stop*, *pause* and *unpause* actions.

This architecture addresses these issues by means of a collaborative service oriented system in which different kinds of services are implemented:

- physical resources services: Hardware Deployment System (Hades), PlatForm Controller (PFC), Resource Manager Service, Machine Manager Service.
- virtual resources services: Virtual Machine Host (VMH), Virtual Machine Manager Service, Virtual Resource Manager Service.
- virtual network services: Vlan Manager Service, Switch Manager Service.
- support services: Account Service, Notifier Service (N).
- user service: User Interface (UI), Client Service.

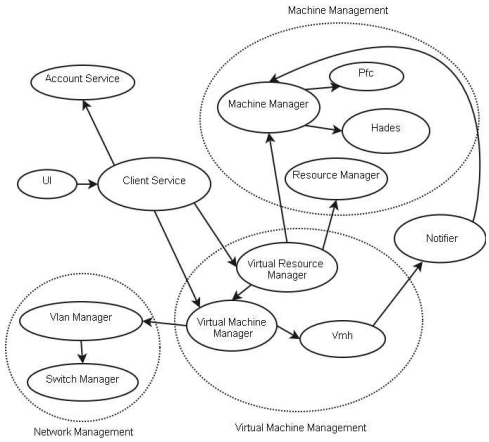


Fig. 2: The architecture's main components.

All architectural components expose their functionalities through a service interface, thus participating in the creation of a SOA-based architecture.

As far as the physical resources are concerned, Hades deals with the deployment of the physical images whereas PFC allows for remotely booting the machines via IPMI by means of power on/off operations. Virtual resources are managed by the VMH service which is replicated in each physical machine; VMH provides operations for creating, starting, pausing and stopping virtual machines on a given physical machine. The Switch Manager is the component responsible for the management of the network devices. Its primary role is to create and maintain the vlans usefull for the communication among the virtual machines. The Machine Manager, the Virtual Machine Manager, the Notifier and the Vlan Manager assemble other services in a workflow structure in order to provide creation of physical and virtual resources.

Finally, Account Service allows for the authentication of users, the Resource Manager as well as the Virtual Resource Manager provide

functionalities for physical as well as virtual machine reservation and booking, and the User Interface gives the user a graphical interface for interoperating with the architecture. It is worth noting that the User Interface and the Client Service must be considered together as the client application. By following a pure service oriented approach, in fact, the user interface becomes a service which only provides the graphical representation of the client application whose logic is coded within the Client Service. In the following we refer to both of them as User Application.

In order to clarify the relationship between the services in the architecture, let us consider all the necessary steps for creating a virtual machine.

1) First of all, the user must log on to the user interface by inserting his username and password. The User Application requests for an authentication to the Account service. If the authentication is verified the user can access the architecture's functionalities.
2) The user inserts the required virtual machines description into the User Interface and then requests its allocation. The User Interface sends a request to the Virtual Resource Manager which calculate the amount of physical nodes needed in order to support the user's request. Then it asks the Resource Manager service for retrieving the free machines to allocate. If the requested machine are available, the Resource Manager returns a *reservation ticket*.
3) The reservation ticket is then used for opening a session on the Virtual Resource Manager that will store all the virtual machine informations for the physical resources allocated within that ticket. The Virtual Resource Manager initializes its inner database by invoking the resource manager for retrieving all the physical resources related to the reservation ticket. From now on, every operation performed by the Virtual Resource Manager on the virtual cluster will be managed through the Virtual Machine Manager.
4) The Virtual Resource Manager asks the Vlan Manager to create a new vlan for the communication among the Virtual Machines. It sends all the hostname of the physical nodes associated with the ticket reservation.
5) The Vlan Manager retrieves the informations regarding the physical topology of the network from its inner database. For each physical node, it finds the switch and the relative port on which the node is connected and then, it asks the Switch Manager to create a new vlan with those informations.
6) The Virtual Resource Manager initializes every selected machine by specifying the setup information. For each of them an initialization request is sent to the Virtual Machine Manager which forwards the request to the Machine Manager. Concurrently, since the initialization procedure could take several minutes, the Virtual Machine Manager subscribes itself to the Notifier in order to be notified when the VMH for the new machine is ready to be used. In the meanwhile the user can perform other operations on the cluster.
7) The Machine Manager starts the initialization for the given machine by invoking Hades, then it sends a request to Hades for the IPMI address of the given machine and, with that information, sends a request to the PFC service in order to reboot the machine.
8) At reboot the machine sends a request to Hades for the image that needs to be installed, downloads it and then installs it. The installed image contains the VMH for that machine which sends a ready notification to the Notifier when started.
9) The Notifier matches the received VMH notification with the previous Virtual Machine Manager request and then sends back a notification to it.
10) The Virtual Machine Manager updates the

state of the physical machine and then forwards the notification to the Virtual Resource Manager.

11) Now, the Virtual Resource Manager can install virtual machines on top of the physical one. It specifies all virtual machine information and then sends a request to the Virtual Machine Manager which will call the related VMH for creating the virtual machine.
12) When the virtual machine is created the VMH sends back an identifier for it to the Virtual Machine Manager that updates its inner database and then notifies the Virtual Resource Manager that the new virtual machine is available.
13) The user can now operate on the virtual machines by starting, stopping or pausing them.

IV. Technologies

In this architecture, the basic features of the SOA approach have been exploited in order to integrate different technologies for implementing the involved services depending on the functionalities they have to provide.

Since Hades, PFC and VMH deal with machine images deployment, we have chosen the Gentoo Linux distribution [7] as the operating system for the machines which will run that services. Then, in order to easily exploit Python Gentoo system utilities, Hades, PFC and VMH have been implemented with the pyGridWare libraries [8] which allow for the management of Web Services SOAP and WSDL specifications in Python. For the workflow services (Machine Manager, Virtual Machine Manager, Virtual Resource Manager, VLan Manager, Client Service and Notifier) we chose JOLIE, which provides easy non-XML primitives for managing incoming and outcoming data, sending and receiving requests and correlating sessions. Finally, since the Account Service, the Resource Manager, the Image Repository, the Switch Manager and the User Interface will be employed in several architectures other than CyberSar, they have been implemented in Java, a well known technology which allows for an easier service management by different developers not directly involved with CyberSar.

V. Conclusion

We have described our strategy for the creation of virtual computer clusters based on standard SOA and hosts virtualization technologies and our ongoing work. Our deployment mechanism, as well as the system management, is delegated to a control plane based on workflows of coordinated web services. The control plane is based on two logically independent modules, the first is responsible of the physical resources and the deployment on the hardware of virtual Xen hypervisor images, while the second manages operations on virtual clusters such as their creation, startup and control. Low level operations e.g., the control of a running image on a given computational host are directly provided by atomic web services running in the dom0 of each participating physical Xen host, while all logic above that level is implemented as Jolie scripts.

References

[1] T. C. Shan and W. W. Hua, "Service-oriented computing kit," *Services Computing, 2006. SCC '06. IEEE International Conference on*, pp. 509–509, Sept. 2006.

[2] L. Sang-Ho, "Ibm grid business model," *Grid Economics and Business Models, 2004. GECON 2004. 1st IEEE International Workshop on*, pp. 145–156, 23 April 2004.

[3] S. Padmanabhuni, J. Ganesh, and D. Moitra, "Web services, grid computing, and business process management: Exploiting complementarities for business agility," in *ICWS '04: Proceedings of the IEEE International Conference on Web Services*. Washington, DC, USA: IEEE Computer Society, 2004, p. 666.

[4] P. Anedda, S. Manca, M. Gaggero, and G. Zanetti, "SOA Based Control Plane for Virtual Clusters," in *Euro-Par 2007 Workshops Parallel Processing - HPPC 2007, UNICORE Summit 2007, and VHPC 2007*, ser. Lecture Notes in Computer Science, vol. 4854. Springer-Verlag, 2007, pp. 154–163.

[5] F. Cappello, E. Caron, M. Dayde, F. Desprez, E. Jeannot, Y. Jegou, S. Lanteri, J. Leduc, N. Melab, G. Mornet, R. Namyst, P. Primet, and O. Richard, "Grid'5000: a large scale, reconfigurable, controlable and monitorable Grid platform," in *Grid'2005 Workshop*. Seattle, USA: IEEE/ACM, November 13-14 2005.

[6] X. Zhang, K. Keahey, I. Foster, and T. Freeman, "Virtual cluster workspaces for grid applications." in *Cluster Computing and the Grid, 2006. CCGRID 06. Sixth IEEE International Symposium on*, 2006.
[7] "Gentoo linux distribution," http://www.gentoo.org/.
[8] J. Boverhof, "pygridware: Python web services resource framework," http://dsd.lbl.gov/gtg/projects/pyGridWare/.

Software Tools for Monitoring ENEA GRID with CRESCO Cluster Integration

A. Funel[1], G. Guarnieri[1], S. Raia[1]
[1]*ENEA, Via Vecchio Macello - 80055 Portici (Naples, Italy)*
E-mail: agostino.funel@enea.it, guido.guarnieri@enea.it, salvatore.raia@enea.it

***Abstract*—The purpose of a Grid network is to collect a large variety of computational resources, often belonging to different architectures, in order to make them a common computational environment with high performances. However, due to its heterogeneity, ENEA GRID requires a careful planning in using available resources and sharing elaborations. For such a structure it is important both to test the network performances and authentication procedures. We present some of the developed tools: the RSH, the UDP and the TCP/IP bandwidth tests. The former is an example of a software which is able to test the efficiency of a command over the full Grid, the second measures the rate at which data can be transmitted over the internet, the third measures the TCP/IP bandwidth. As CRESCO has been fully integrated in ENEA GRID we have developed a tool for monitoring the status of some of its essential functionalities: the INFINIBAND connections, the memory status, the SSH protocol, the GPFS parallel file system and scheduler LSF. ENEA Grid and CRESCO have a wide range of compilers and parallel libraries used in parallel applications. There are difficulties to select and set the suitable environment: processes and threads spawning for hybrid parallel applications, resources requirement in LSF framework, queues selection as well as code integration with LSF and communication protocols architecture. We have developed a software consisting of a set of Parallel Job Launcher (PJL) and wrappers for submitting jobs to LSF. PJLs allow integration between codes and LSF for the main parallel libraries, while LSF wrappers allow users to see a single environment without warring about the choice of compilers and launchers. Finally, we use the above tools to measure the computational power of CRESCO HPC system and how to vary parameters such as number of processes, threads and processes per host. We take as application examples CPMD (CP Molecular Dynamics) and HPL (High Performance Linpack).**

I. NETWORKING

ENEA GRID machines are located in six ENEA Centers geographically distributed over Italy. Even if the AFS file system prevents the necessity of sending very large data, many applications will work fine only if the network is fully efficient. The robustness of the network is especially important when dealing with sockets as in the case of parallel codes implemented according to the message passing paradigm. The tools which will be presented aim to test the performances of the two main communication protocols, the TCP/IP and the UDP.

A. TCP/IP BANDWIDTH TEST

This tool measures the TCP/IP [1] bandwidth between two hosts of ENEA GRID. In this way it is possible to detect bottlenecks which may cause the traffic to slow down or, badly, the failure of applications with a low time out threshold. The implemented code is written in C and follows the client-server scheme as shown in Fig. 1. The master process relays on the scheduler (LSF) [2] to start the client and server. The user can set the range of the band to analyse. Each data packet travels many times between client and server to smooth out sudden

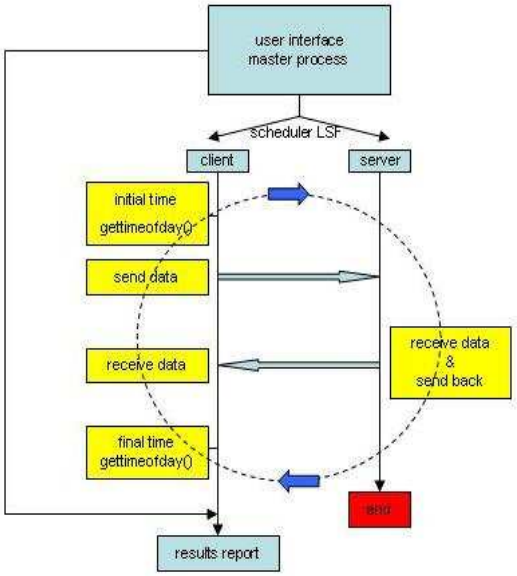


Fig. 1. Flow chart of the TCP/IB Bandwidth Test for ENEA GRID.

falls of the network which may be always present but do not affect the average performance. Results are reported both in textual ang graphical fashion for an easy consultancy.

B. UDP TEST

The aim of the test is to check the shape of the network response time between two hosts (client and server) of ENEA GRID. We define the network response time as the round trip time it takes to send a packet of data and receive it back. Thus it is a measure of the communication speed between two remote hosts. The client sends a packet to the server and once the server receives it, the packet is immediately sent back to the client. The test uses the UDP protocol [1]. UDP protocol does not have any internal acknowledgment procedure to ensure reliability, ordering or data integrity. It is used in time sensitive applications which do not require error checking and correction. UDP protocol is faster than TCP/IP. The user can set the time step between the sendings of data packets. As the overhead is reduced to the minimun the test can be used to study the velocity of data transmition as a function of time and data size.

II. MONITORING

ENEA GRID is a heterogeneous environment in that many different operating systems and architectures are present. The necessisy of sharing common resources is a major issue which requires a carefull monitoring. Especially important is to take under control those procedures which allow remote access for launching commands and running applications. However, for a local cluster as CRESCO it is crucial to ensure full efficiency for its own functionalities (like, for example, the connection between processor nodes). The former of the two tools we shall present, the *RSH TEST*, focuses on global monitoring while the latter, *CRESCO MONITORING*, is specifically designed for the new CRESCO cluster.

A. RSH TEST

The `rsh` (remote shell) program executes shell commands on a remote computer. It uses the network to connect and authenticate on a remote host. Even if the protocol is not secure because data are not encrypted, it is often used as communication agent by parallel applications because unencrypted data can be quickly transmitted. The developed tool checks which hosts can be reached via `rsh` from a starting node of ENEA GRID. This one-to-all statistics can be easily extended over the totality of the machines. As we use the LSF scheduler for running the test on the starting node independently on the host the user is logged on, the tool provides also an indirect monitoring of the scheduler. The structure of the developed code, illustrated in Fig. 2, is very flexible because such scheme can be modified with minor effort for monitoring all network based protocols.

B. CRESCO MONITORING

This is a tool specifically designed for a fast and efficient diagnostic of the main functionalities of the CRESCO cluster [3]. Many things

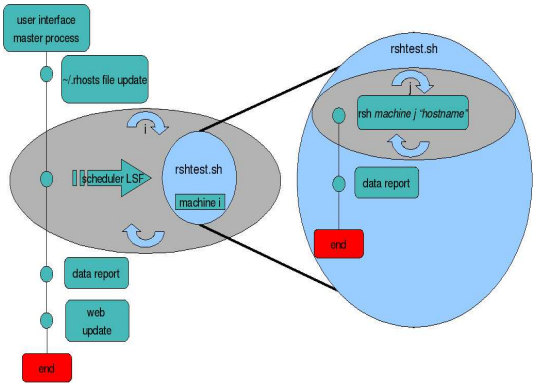


Fig. 2. RSH TEST algorithm flow chart. As first thing the `.rhosts` file is updated in order to correctly set permissions. There are two nested loops. The first loop is over the set of the selected machines. By means of the scheduler (LSF) a script (`rshtest.sh`) is started for each of them. Such script performs the second loop in which all remote accesses via `rsh` are tested. At the end all data are reported both in a local user directory and on the web.

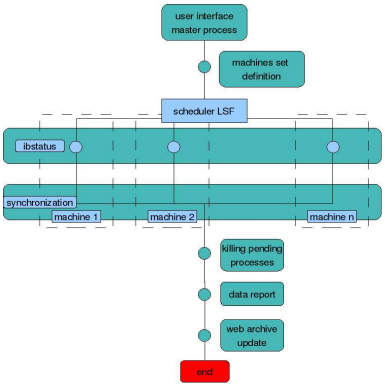


Fig. 3. Flow chart of the algorithm used by *CRESCO MONITORING* for checking the InfiniBand (IB) status. The same technique is used for monitoring the RAM memory status and scheduler (LSF) demons. Results are reported both in a local user directory and on the web.

need to be working properly for CRESCO being fully operative, like network connections, communication protocols (SSH etc.), the scheduler (LSF) etc. *CRESCO MONITORING* is a set of programs each of which can be easily started from command line. CRESCO computational nodes are divided in two sections, one section consists of 42 nodes and is dedicated to jobs which require a lot of memory, the other one consists of 256 nodes and is dedicated to massive parallel jobs. *CRESCO MONITORING* allows users to choose which portion of the cluster to analyse with the possibility of selecting the totality of the machines. CRESCO is equipped with InfiniBand (IB), a switched fabric communication link for high speed interconnection between processors [4]. *CRESCO MONITORING* can analyse the IB status for all CRESCO machines in a few seconds. This is accomplished by launching sequenzially in background both via scheduler (LSF) and via SSH (secure shell protocol) [5] a shell command which for each node tests the status of the IB board. All these background processes are synchronized within a loop which waits until the last test is done. All zombie processes or processes which exceed the waiting time threshold are killed. The algorithm is illustrated in Fig. 3.

This technique has the extra advantage of keeping track of those nodes which may have problems with the scheduler. All codes running on CRESCO are managed by the scheduler (LSF) which role is to correctly employ the available resources. *CRESCO MONITORING* can quickly detect, for all CRESCO nodes, critical situations of the scheduler. This task is accomplished by checking whether all processes (demons) which permit the scheduler to work correctly, are operative. In the same fashion *CRESCO MONITORING* performs the task of checking the RAM memory status.

Another feature of *CRESCO MONITORING* is to check whether SSH works from a given set (say A) of CRESCO machines to another set (say B) of CRESCO machines. In this case the technique is more involved. In fact, an SSH server can not hold concurrently more than a given number of connections. If a host attempts to connect to a busy SSH server the connection will fail but it does not mean that there is a problem with SSH. At a later time, when the server will be ready to accept a connection, a new attempt will succed. To prevent from reporting not existing errors a new algorithm

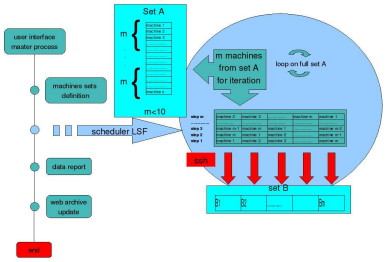


Fig. 4. Flow chart of the algorithm used by *CRESCO MONITORING* for checking SSH. The set *A* of machines from which one wants to test the SSH protocol is chunked in blocks which size *m* is at most equal to the maximum number of concurrently allowed connections for an SSH server. The set *B* of machines towards which one wants to access via SSH is divided in *m* subsets *B1,...,Bm*. A loop is performed over set *A*. For each iteration *m* machines belonging to set *A* are selected then a second loop of *m* steps is started. For each step of the second loop there is a one to one correspondence between the selected *m* machines of set *A* and the *m* subsets *B1,...,Bm*. During the second loop a cyclic permutation fulfils all possible correspondences. Results are reported both in a local user directory and on the web.

was designed. Each SSH server belonging to set B was not allowed to accept any further connection from a client of set A until all the previous requests were satisfied. Once a server become free a new group of clients from set A fulfils all the available connections. This chain works in a pipeline way to reduce dead times and avoid overlaps. The algorithm is outlined in Fig. 4.

III. MPI-OpenMP Application Launchers and LSF Integration

Launching parallel (MPI [6], OpenMP [7] and hybrid codes) applications under LSF requires users to do the following steps: environment settings for MPI libraries; span of processes, threads and processes per host (task locality specification); selection of PJL (Parallel Job Launchers) wrappers for OpenMPI [8] and MVAPICH [9], queues and other standard LSF options.

We have developed a set of programs that perform the above tasks, simply by specifying

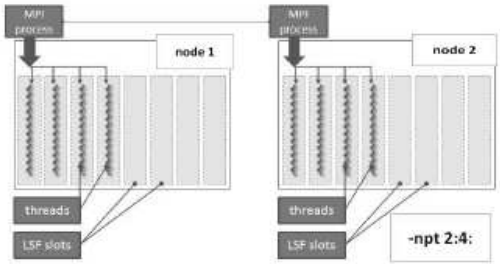


Fig. 5. Processes spanned over two hosts through *-npt 2:4:1* string.

the following string to the main program, called **mpi_omp.bsub**:

-npt x:y:z

where *x* specifies the number of processes, *y* the number of threads that each MPI process have to span and *z* the number of processes that LSF should launch on each host. It is a general tool that allows to submit any kind of parallel job. The following subsections and figures show how the tool works and some examples.

A. Setting the environment for MPI libraries

On CRESCO cluster three main MPI implementations are available: OpenMPI, MVAPICH, LAM/MPI [10]. Furthermore, for each MPI library, there are the following programming environments, i.e. compilers, mathematical and numerical libraries etc.: gnu, intel and pgi. In order to correctly select the environment before launching parallel MPI applications, configuration files have to be sourced. Our tool checks the current MPI library selection and source the suitable system scripts, leaving the user free of dealing with them.

B. Parallel task locality specification

Hybrid programming models use different parallel paradigms at different level of parallelism. For example, in clusters with distributed shared memory processors (SMP) nodes, one could use OpenMP (Open specification for Multi Processing) for the fine-grained parallel

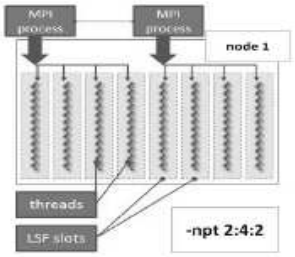


Fig. 6. Processes spanned over a single host through *-npt 2:4:2* string.

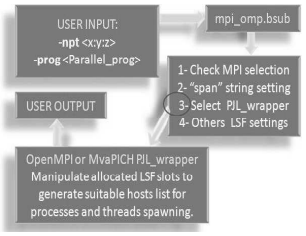


Fig. 7. In this schema the selected PJL wrapper have to mark the difference between processes and threads.

CASE: 3 diamond supercell, 512 atoms			
Tot. tasks	-npt	Parallel type	Iter/sec
256	32:8:1	MPI+OpenMP (1 process/host)	0.028
	64:4:1	MPI+OpenMP (1 process/host)	0.026
	64:4:2	MPI+OpenMP (2 process/host)	0.037
	128:2:1	MPI+OpenMP (1 process/host)	0.017
	128:2:2	MPI+OpenMP (2 process/host)	0.032
	128:2:4	MPI+OpenMP (4 process/host)	0.036
	256:1:2	Only MPI (2 process/host)	0.040
	256:1:4	Only MPI (4 process/host)	0.035
	256:1:8	Only MPI (8 process/host)	0.019

Fig. 8. Processes span on two hosts through *-npt 2:4:1* string.

tasks assignment and MPI (Message Passing Interface) for the coarse-grained parallelism. Despite MPI implementations allow users to set the number of processes and define environment variables for specifying how many threads each process has to span, this is quite complicated with the scheduler LSF. Parallel tasks locality specification under LSF submission is done via the *span* string in the resources requirement. Our tool set LSF resources requirement and tasks locality via the *-npt* option. CRESCO cluster provides the possibility to accomodate processes and threads in many combinations because it is constituted of 42 SMP nodes with 16 cores each and 256 SMP nodes with 8 cores. For a user, it is important to have the chance to tune its own code in order to reach the best combination. Fig. 5 and Fig. 6 show as an example the following case: 8 total parallel tasks, 2 MPI processes and 4 OpenMP threads per process. There are two way, in this case, of spanning MPI processes on each host: 1 processe per host (*-npt 8:4:1*) taking 2 hosts or 2 processes per host (*-npt 8:4:2*) taking 1 host. Here we suppose that each node has 8 LSF slots. As we will show in the section dedicated to CPMD (Carr-Parrinello Molecular Dynamics) code [11], such allocation task affects the performance of codes.

C. PJL wrapper selection

MPI and/or OpenMP jobs integration with LSF is accomplished via PJL wrappers. On CRESCO cluster such mechanism uses the LSF *blaunch* framework. There are two problems that our tool solves for hybrid codes: hot to correctly utilise the right PJL wrapper (Open-MPI or MVAPICH) and how to create LSF hostfiles. In fact, the vendor PJL (i.e. mpirun) needs to know the hosts on which to launch MPI processes, but LSF just allows to allocate slots according to the *span* string. It does not make any difference between processes and threads, and this must be controlled by the user who have to create an hostfile. Fig. 7 shows the schema followed by *mpi_omp.bsub*.

IV. CPMD TESTS USING *mpi_omp.bsub* PROGRAM

In this section we show an example of utilization of the *mpi_omp.bsub* program in order to tuning or analyse how hybrid programming can improve performances. We take CPMD (Car-Parrinello Molecular Dynamics) code and let vary the *-npt* parameter over 256 SMP nodes Intel Xeon (8 cores per node). Fig. 8 shows a test where the total number of parallel tasks is kept constant, and the *-npt* parameter is

changed in order to get the best combination. Last column represents the iterations per second in a molecular dynamics simulation. Our system consists of 3 diamond supercells of 512 atoms. The case *npt 256:1:8* corresponds to the usual way of submitting jobs, i.e. only MPI processes filling all cores/slots on each host.
Our test makes it clear that the best performance, in this case, is achieved by launching only 2 MPI processes per host without threads. In this configuration the job runs across 128 hosts using only 2 of the 8 LSF slots per host. Clearly, the free slots could be allocated for other jobs and this eventuality may degrade the performance of the code (because, for example, the total memory will be shared with the new jobs). However, the best way of submitting hybrid codes is by filling all the available slots on each host. We have found that the best ratio processes/threads, that satisfies this requirement, is the 2/4 or 4/2. We get, respectly, 0.036 and 0.037 dynamic molecular iterations per second, good values compared with the best value of 0.040.

V. Conclusion

We have developed software tools for the management of ENEA GRID with CRESCO cluster integration. The main purpose of our work has been to provide the user the manner of using the many available different architectures comfortably. For this reason we have focused both on global and local issues. While the access to the resources distributed over the six ENEA Centers requires an efficient monitoring of the network, the code optimization on a local cluster of ENEA GRID, like CRESCO, is not possible without tools specifically dedicated to job submition and environment settings. Our tools attempt to accomplish these two tasks.

Acknowledgments

We wish to tank Silvio Migliori, Salvatore Podda, Giovanni Bracco and the stuff of the ENEA-Portici Research Center for their support.

References

[1] W. R. Stevens *UNIX Network Programming*, Vol. 1 and 2, 2nd ed. Prentice Hall, 1999.
[2] http://www.platform.com/Products/platform-lsf
[3] http://www.cresco.enea.it/
[4] http://www.infinibandta.org/home
[5] http://www.openssh.org/
[6] http://www-unix.mcs.anl.gov/mpi/
[7] http://openmp.org/wp/
[8] http://www.open-mpi.org/
[9] http://mvapich.cse.ohio-state.edu/
[10] http://www.lam-mpi.org/
[11] http://www.cpmd.org/

Web Crawling on ENEA GRID infrastructure to perform Web Archive

S. Raia[1], S. Migliori[2], M. Santini[3], D. Giammattei[1], M. De Rosa[1]

[1]*ENEA, Portici Research Center, Piazzale E. Fermi, 1 - 80055 Portici, Naples, Italy*
[2]*ENEA, Via lungotevere Thaon di Revel, 76, Rome, Italy*
[3]*Computer Science Department, University of Milan, V. Comelico 39, Milano, Italy*

***Abstract*—A web crawler automates the process of following links through the Internet and creates a copy of all visited pages. Large scale crawling presents several performances issues. In this paper we show how to overcome them, by setting and tuning a parallel Web Crawler on ENEA Grid.**

I. Introduction

The explosive growth of the web represents one of the main challenge for web search engines in order to index and retrieve its contents. Understanding web structure, is also becoming of great interest for social and commercial purposes. Prerequisite for studying and analysing the web is to collect it. Web Crawling allows to gather as many useful web pages as possible, together with the link structure that interconnects them. Due to the large size, web sites may have significant changes before the crawling process finish. For that reason Web Crawlers software should be very efficient and have a software architecture that allows to use computational resources to perform tasks in parallel and/or distributed environment. Our goal is to make snapshots of significant portions of the web, e.g. '.it' domain, consisting of about hundreds of millions of pages, as more efficiently as possible. For that we have optimized and integrated the UbiCrawler application in ENEA Grid. UbiCrawler is a scalable distributed crawler, fault tolerant and platform independent. It is written in Java language. Crawling significant portions of the web some performance related problems come out: available bandwidth toward the web,

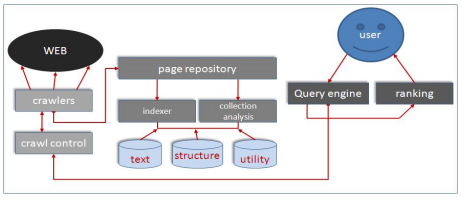


Fig. 1. A schematic representation of search engine logical operation.

efficiency of file systems to which I/O operation are performed and load balancing both during the crawling and for post processing. We show some results about integration of UbiCrawler on ENEA Grid, that allows us to exploit the effective available bandwidth on CRESCO computational centre and the other sites distributed in ENEA Grid. The file system bottleneck is overcome using parallel file system GPFS and distributing volumes across AFS. We also present some tools and utilities used to make post processing operations on the crawled web sites. Since data gathered from web are quite huge (of Tera Byte order) and split into different many pieces, we use the functionality of LSF framework to exploit CRESCO resources and minimize time execution for such processes.

II. Web Crawling and Web Structure

Web Crawling is a prerequisite to study and analyze web structure. The web is dynamic and knowledge of its evolution is increasingly being used to improve organization, search, and

analysis of information on the web. Web structure evolution determine web algorithms for crawling, searching and community discovery and sociological phenomena, but these in turn determine web evolution. It is a self consistent system.

Web crawlers work in the following way: start by parsing a specified set of web pages, noting any hypertext links on that pages that point to other web pages. They then parse those pages for new links, and so on, recursively. A crawler resides on a single or a cluster machines and sends HTTP requests for documents to other hosts on the Internet. All the crawler really does is to automate the process of following links. Fig. 1 shows an example of crawlers use: the mechanism underlying a Web search engine. Crawling large portions of the web, following main problems come out:

- available bandwidth toward the web is limited for parallel or distributed crawling;
- performance of filesystem is critical, because of high number of I/O operations during the crawling;
- dimension of stored web size is huge for large scale crawling;

After a description of UbiCrawler, the application we used to make snapshots of '.it' domain and crawling experiments, we describe how to cope with the above limitations on ENEA Grid infrastructure.

III. UBI CRAWLER SOFTWARE ARCHITECTURE

In ENEA Grid we use the crawler software, called *UbiCrawler*, to collect data from the web and perform web archive. Follow a brief description of its general design and implementation [1]. UbiCrawler is a scalable, fault tolerant and fully distributed web crawler. Main features of UbiCrawler are:

- platform independence: this is achieved because implementation language used is Java;

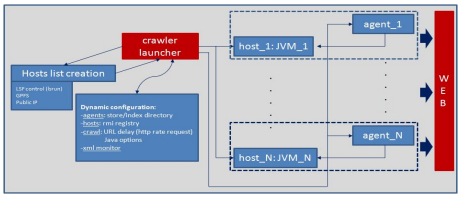


Fig. 2. Structure of UbiCrawler launching process.

- full distribution of every parallel task: no single point of failure and no centralized coordination at all;
- locally computable URL assignment based on consistent hashing;
- tolerance to failures;
- scalability.

UbiCrawler is composed by several agents that autonomously scans the assigned portion of the web space. An agent performs its task by running several threads, each dedicated to the visit of a single host. Each thread scans a single host using a breadth-first visit. The outlinks that are not local to the given host are dispatched to the right agent, which puts them in the queue of pages to be visited. The choice of Java as implementation language allows to achieve platform independence. Speed of UbiCrawler is limited by network bandwidth and memory, and not by CPU power. Java made possible to adopt Remote Method Invocation (RMI), a technology that enables one to create distributed applications in which the methods of remote Java objects can be invoked from other Java virtual machines.

IV. UBICRAWLER INTEGRATION ON ENEA GRID NETWORK

UbiCrawler architecture is complex and to set a crawler session, following steps are required:

1) setting configuration files:
   - hosts: here we place hostnames, RMI registry names and Java options;
   - agents: specification of agent capacity, how many threads to span and the directories where save index, store and log files;

- domain to crawl: it is possible to set the max depth in the recursive link visit, as well as a parameter that set the interval between two http requests;
- monitor: specify the host where web monitor will run;

2) bootstrap of Java Virtual Machine (JVM) for given hosts;
3) agents creation;
4) dispatch of a set of starting URLs;
5) monitoring the crawlers;

We have set a launcher that accepts a hosts list and does the above work. A host is added to the list if it is available via LSF and mounts GPFS (General Parallel File System) filesystem, in order to allows thousands of threads write log files and store web pages as efficient as possible. The Crawler launcher bootstraps JVM via LSF program 'lsrun'. We choose LSF tools to execute remote start up commands, in order to achieve LSF integration of the software and make easier batch submission. Each JVM can house one or more agents. We run agents on CRESCO cluster, consisting of *smp* Intel Xeon nodes with 8 cores [5]. Every agent creates a team of threads, so it is possible to exploit the multiprocessor architectures increasing the number of threads and/or agents. Beside basic settings written in the configuration files, in the first step, one can decide the web domain to crawl and the number of agents per host. At run time it is also possible to change the frequency to which a thread does http requests to the web hosts. This point is critical since crawlers saturate network bandwidth and controlling such parameter allow us the best plan for long time crawls.

V. High Performance of Large Scale Crawling on ENEA Grid Network

Network bandwidth is the main limit to improve performance, as stressed in previous sections. Here, we shows some experimental and complete crawl results. Fig. 3 and fig. 4 represent the network bandwidth occupied during a crawl experiments. We let vary the number

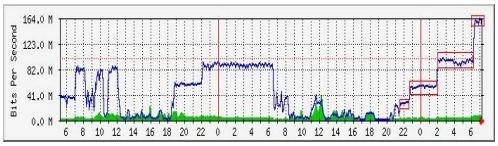


Fig. 3. In this experiment we use 4 agents on Cresco clusters. The stair-steps on the right of picture, correspond to double the value of http request rate (URL delay). Graph values are 5 minute average and expressed in Mega bits per seconds (Mbps).

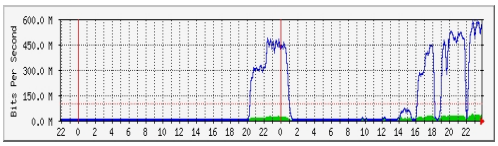


Fig. 4. Same crawl tests as in Fig. 3, but we scale the number of agents until we reach the maximum of about 650 Mbps.

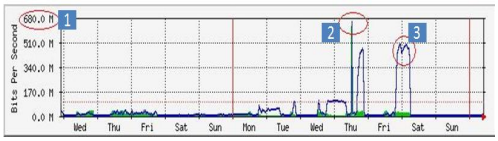


Fig. 5. Graph values are 30 minute average, expressed in Mbps. Red circle 1 and 2 is the network peak performance obtained during a crawl with 20 agents running on CRESCO Portici site. Circle 3 represent a crawl of '.it' domain run for 8 hours (8,75 million pages/hour).

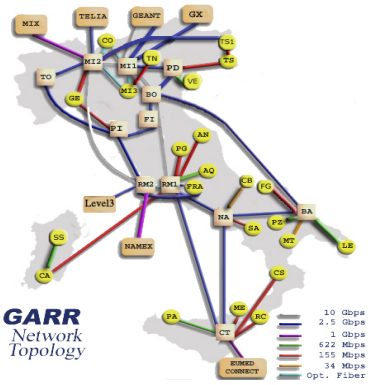


Fig. 6. ENEA computational resources are distributed over WAN, connected by GARR, the Italian Academic & Research Network.

of agents and a parameter, called 'URL delay', that set the rate which any thread send http requests to web hosts. While the number of agents is arranged at bootstrap, http rate request parameter can be changed at run time. Units are one http request per milliseconds: e.g. 250 means 1 http request every 250 ms (i.e. 4 pages/sec), 500 means 1 http request every 500 ms (2 pages/sec) and so on.

We can take snapshots of sizeable portions of the web, in reasonable time, using about 20 agents with the above mentioned requisites. Fig. 5 shows a crawl of '.it' domain: we have harvested about 80 million of pages, which data stores are about 500 GB. The elapsed time was 8 hours. With the same performances we can make snapshots up to 100-120 million of pages.

Others snapshots of the web were taken spanning the agents across different hosts on ENEA Grid (see Fig. 6). This allows us to increase the aggregate available bandwidth. In a crawl experiment we run agents on three different cluster of ENEA Grid. Using the same parameters as the crawl described in Fig. 3, we start 15 agents on Portici site, 4 agents on Frascati site and 1 agent on Brindisi site. The max available bandwidth for users on Portici PoP GARR site link is 650 Mbps. The measured bandwidth on Frascati and Brindisi PoP Garr link was 20 Mbps/agents [4]. Tuning the number of agents it is possible to reach an effective total bandwidth of 1Gbps across the ENEA Grid network. This setting allows to increase the rate of pages saved per hour from 8.75 Million/hour up to 10 Million/hour.

VI. Web Archive in WARC format

Web pages gathered during crawling are stored in WARC (Web ARChive) format [3]. Each agent will write data in one store file. Large scale crawling, using 20-25 agents, generate few large file of 50-60 GB. We combine LSF framework utilities and LAW (Laboratory for Web Algorithms) [2] packages to manage and retrieve information from WARC files. So we could recreate locally the web structure of a given domain, as well as study the web graph or retrieve information from our archives.

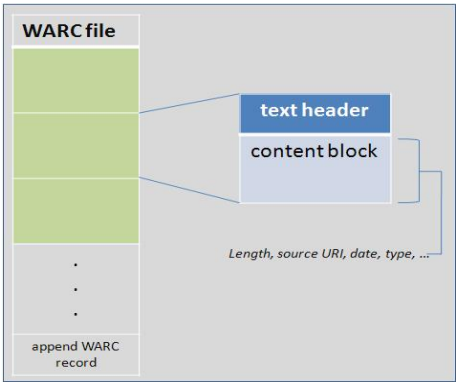


Fig. 7. WARC file is constituted of WARC records. WARC header identify information about the current record, and allows additional per-record information. Each record's content block contains zero or more bytes of data, interpreted according to the record type and any preceding headers.

VII. Conclusion

As shown in this paper, large scale web crawling demands parallel or distributed web crawlers software as well as meaningful computational resources. Integration and tuning of UbiCrawler application in ENEA Grid allows to overcome network bandwidth and filesystem bottlenecks. Spreading agent crawlers across three computational sites on ENEA Grid, we can scale up to 1Gbps of effective network bandwidth. Our next step is to exploit all of the six computational centres and possibly other Grids, so that we can improve crawling performances and the size of web domains to crawl.

References

[1] P. Boldi, B. Codenotti, M. Santini, and S. Vigna. *Ubicrawler: A scalable fully distributed web crawler*, Software: Practice & Experience, 34(8): 711-726, 2004.
[2] *High-performance parallel web crawling*, http://law.dsi.unimi.it/
[3] *ISO/DIS 28500: Information and documentation. The WARC File Format.*, http://bibnum.bnf.fr/WARC/warc_ISO_DIS_28500.pdf
[4] *GARR consortium: La rete Italiana dellUniversit e della Ricerca*, http://www.garr.it
[5] *CRESCO: Centro Computazionale di Ricerca sui Sistemi Complessi*, http://www.cresco.enea.it

Multilevel Algorithms for Multidimensional Integration in High Performance Computing Environment

Giuliano Laccetti[1], Marco Lapegna[1], Diego Romano[1] and Valeria Mele[1]

[1]*Department of Mathematics and Applications, University of Naples Federico II*
via Cintia Monte S.Angelo – 80126 Naples (Italy)

***Abstract*— A multilevel parallel adaptive algorithm for the approximate computation of a multidimensional integral over an hyperrectangular region is described. This new algorithm is aimed at an efficient implementation on a MIMD distributed memory Multiprocessor with multi-core CPUs. For this purpose two levels have been introduced in the algorithm: an upper level is responsible of the communication among the nodes and a lower level is responsible of the coordination of the cores in a single node. First performance results on a Blade Server with 16 nodes and 2 quad-core CPUs per node have been achieved.**

I. INTRODUCTION

FROM an architectural point of view, a High Performance Computing environment can be described by means of a tree structure, where at the lowest level there are multi-core CPUs, at the middle level there are workstation clusters and blade servers, and at the highest level the wide area network connects the systems. For such a reason, the development of algorithms and scientific software for these environments needs the integration of several methodologies to face the different kinds of parallelism of each architectural level (distribution of the computation among the systems, parallelism among the CPUs as well among the cores of a single CPU). The aim of this work is to show the experiences gained during the development of adaptive algorithms for multidimensional integration on high performance computing systems (cluster of workstations, MPP systems, blade servers) with multi-core CPUs. In the proposed algorithm the tasks are dynamically scheduled on the cores of a single CPUs accordingly to the availability of the local resources, and periodically rearranged among the CPUs of the system to balance the workload. This two-levels approach produces an algorithm with a reduced number of synchronization points among the tasks and, at the same time, a high degree of scalability.

II. ADAPTIVE ALGORITHMS FOR MULTIDIMENSIONAL INTEGRATION

Since the scientific importance of the numerical computation of a multidimensional integral:

$$I(f) = \int_U f(t_1, \ldots t_n) dt_1 \ldots dt_n \qquad (1)$$

in the past several efficient routines for the solution of this problem have been developed, where $U = [a_1, b_1] \times \ldots \times [a_n, b_n]$ is a *n*-dimensional hyper-rectangular region. Most of them (see for example [1, 6, 8]) are based on adaptive algorithms that allow high accuracy with a reasonable computational cost.

Such algorithms are iterative procedure that, at each iteration *j*, evaluates a composite integration rule on a family of sub-domains $s_k^{(j)}$ *(k=1,..,K)* of a partition $\wp^{(j)}$ of *U*. The aim is to compute an approximation *Q(f)* of *I(f)* and an estimate *|E(f)|* of the absolute error *|I(f)-Q(f)|* such that:

$$|I(f)-Q(f)|<|E(f)|<\varepsilon \quad (2)$$

where ε is an user-requested tolerance. To achieve this, the algorithm computes a sequence $Q^{(j)}$ of composite quadrature rules defined on the sub-domains of $\wp^{(j)}$ approaching *I(f)* and a sequence $|E^{(j)}|$ of approximations of the error approaching 0, until the user-requested tolerance is satisfied or it is realized that the error criterion (2) cannot be achieved within a allowed bound on the number of function evaluations. For dimension up to 15 we have good rules available for standard regions [2] .

Since the convergence rate of this procedure depends on the behavior of the integrand function (presence of peaks, oscillations, etc), in order to reduce as soon as possible the error, at the iteration *j* the sub-domain $\hat{s}^{(j-1)} \in \wp^{(j-1)}$ with maximum error estimate $\hat{e}^{(j-1)} = \max_{j=k,..,K} e_k^{(j-1)}$ is split in two (or more) parts s_λ and s_μ that take the place of $\hat{s}^{(j-1)}$ in the partition $\wp^{(j-1)}$. In the same way the approximations $Q^{(j)}$ and $|E^{(j)}|$ are updated.

Usually the sub-domain $\hat{s}^{(j-1)}$ is split in only two parts by halving the edges along the direction where the fourth divided difference of the integrand function is larger than in the other directions. Such a value is taken as a measure of the difficulty of the integration in that direction. This kind of procedure is called *global* adaptive algorithm. A framework for a sequential global adaptive algorithm for the computation of multidimensional integrals is therefore the following :

begin Algorithm 1
Initialize $\wp^{(0)}$, $Q^{(0)}$ and $|E^{(0)}|$
while (stopping criterion not satisfied)
1) find $\hat{s}^{(j-1)} \in \wp^{(j-1)}$ such that $\hat{e}^{(j-1)} = \max_{k=1,..,K} e_k^{(j-1)}$
2) divide $\hat{s}^{(j-1)}$ in two parts s_λ and s_μ
3) evaluate the quadrature rules and compute the error estimates in s_λ and s_μ
4) insert in $\wp^{(j)}$ the sub-domains according their error estimates
5) update $Q^{(j)}$ and $|E^{(j)}|$
endwhile
end Algorithm 1

There are several strategy to introduce parallelism in a quadrature algorithm [7] , but in order to achieve high efficiency our approach is based on the distribution of the sub-domains of $\wp^{(j)}$ among the processing units so that it is possible to process them concurrently. This approach is known as Parallelism at Subdivision Level and allows to introduce a sufficient large granularity in the algorithms with a reduced cost in synchronization and communication . The main problem of this approach is that the sequence of $\wp^{(j)}$ is unpredictable, so that it is impossible to distribute uniformly the workload among the computing units beforehand. Therefore, to ensure proper balancing, a parallel algorithm needs to reorganize periodically a more suitable work subdivision among the computing units of the system (nodes as well as cores into each

node). This type of load balancing (sometime called *dynamic load balancing*) reacts to the current state of the algorithm during the execution of parallel tasks and it can improve the performance of parallel systems in case of unpredictable workload.

With the aim of defining a suitable strategy to redistribute the sub-domains among the processing units, let us observe that the two architectural levels in a MIMD distributed memory system with multi-core CPUs (node level and core level) require different algorithms development methodologies because of the very different architectural features. The upper level require a traditional approach based on message passing with the issue of reducing the communication overhead, whereas the lower level require an approach based on the concurrent access to shared memory with the other issue of reducing the synchronization overhead. For these reasons we propose a multilevel algorithm, where each level is responsible to manage data for the corresponding architectural level.
Three separate discussions on these levels and how they can be combined together in the same algorithm follow.

A. Parallelization among the cores

To implement Algorithm 1 in a multi-core based computing system, firstly we observe that the cores cannot be considered as completely independent processing unit, because they share both on-chip resources like caches or bus, as well external resources like the main memory or I/O devices [5]. These hardware features draw a picture where several computing units access to shared resources, so that a natural programming paradigm for an efficient use of these devices is based on the Thread Level Parallelism. With this approach, *P* threads can execute concurrently the Algorithm 1, with a completely asynchronous and out-of-order execution. However, it should be noted that to achieve high efficiencies in these computing environments, a key issue is the asynchronicity of the tasks accessing the on-chip shared resources. This is a critical issue for an adaptive algorithm like Algorithm 1, because, as we said, such an algorithm have to select the sub-domain with maximum error estimate to reduce as fast as possible the error estimate. Usually in an adaptive algorithm for multidimensional quadrature, the sub-domains are arranged in a data structure that has to be accessed by all threads. For a multicore version of Algorithm 1, the data structure resides in main memory which implies the steps 1) and 4) being two critical sections of the algorithm because of the risk for race condition on this shared data structure. However it should be noticed that the computational cost of such steps executed in critical sections is generally much smaller than the cost of several step 3) executions concurrent on the cores, since there are several data structures with a management computational cost of $O(\log_2 K)$, where K is the number of the stored sub-domains. Such a value is much smaller than the cost for evaluating an integration formula, mainly for large values of *n*. For such a reasons, the idle times are almost completely eliminated in Algorithm 1 when executed on a multi-core CPU.

B. Parallelization among the nodes

After the sub-domains refinement among the cores described in the previous subsection, in order to ensure proper load balancing among the nodes, it is necessary to compare periodically the data in their private memory and to reorganize a more suitable sub-domains distribution in the upper level. A common strategy in this sense is to exchange the sub-domains with a large error among the nodes.

However we remember that, since a necessary condition for the scalability is the data locality, the communication time for the sub-domains rearrangements must be independent on the size of the system (i.e. the number of nodes). So all global communications are removed to improve the scalability of our algorithm, because such communication tasks depend on the number of nodes and make the algorithm poorly scalable. Meanwhile the workload is balanced by performing only communications between pairs of directly connected nodes in a 2-dimensional periodical mesh [3, 4] . In such a topology each node has 4 neighbours: 2 for every direction. In our algorithm each node attempts to send its sub-domain with the largest error estimate to the connected nodes in the mesh, in order to share it with the other nodes. More precisely, let $\hat{e}_i$, $\hat{e}_{i+}$ and $\hat{e}_{i-}$ be respectively the largest error estimates of the sub-domains in the current node P_i, in the next node P_{i+} and in the previous node P_{i-} . If $\hat{e}_i > \hat{e}_{i+}$ then the current node P_i sends the sub-domain with largest error estimate $\hat{s}_i$ to the processor P_{i+} . In the same way if $\hat{e}_{i-} > \hat{e}_i$ the current node receives the sub-domain with error estimate $\hat{e}_{i-}$ from P_{i-} . In such a workload distribution, the communication cost is independent of the number of nodes in the system, thus enhancing the scalability of the algorithm.

C. The multilevel algorithm

At the end of this section we give a description of the whole multilevel algorithm. Algorithm 2 is the node algorithm in the Single Program Multiple Data (SPMD) model for the node P_i. The inner loop is implemented in a thread and replicated on each core of the nodes.

begin Algorithm 2 for node P_i

Initialize $\wp_i^{(0)}$, $Q_i^{(0)}$ and $|E_i^{(0)}|$

while (node stopping criterion not satisfied)

Define the neighbors nodes P_{i+} and P_{i-}

if ($\hat{e}_i > \hat{e}_{i+}$) send $\hat{s}_i$ to P_{i+}

if ($\hat{e}_{i-} > \hat{e}_i$) receive $\hat{s}_i$ from P_{i-}

while (core stopping criterion not satisfied)

{enter critical section}

1) select $\hat{s}_i^{(j-1)} \in \wp_i^{(j-1)}$ with largest error estimate in $\wp_i^{(j-1)}$

{exit critical section}

2) divide $\hat{s}_i^{(j-1)}$ into two parts s_λ and s_μ

3) evaluate the quadrature rules and compute the error estimates in s_λ and s_μ

{enter critical section}

4) insert in $\wp_i^{(j)}$ the sub-domains according to their error estimates

{exit critical section}

5) update $Q_i^{(j)}$ and $|E_i^{(j)}|$

endwhile

endwhile

compute $Q(f) = \sum_i Q_i^{(f)}$ and

$|E(f)| = \sum_i |E_i^{(f)}|$

end Algorithm 2

III. TEST RESULTS

In this section we present some results achieved on a blade server with 16 nodes each

of them equipped with two Intel XEON E5410 quad-core CPU and 16 Gbytes of main memory.

The following Figure 1 refers to the Algorithm 2 implemented on a single node. Such a test is therefore aimed to measure the effectiveness of the lower level of the algorithm, i.e. the level responsible of the dynamic load balancing on the cores in the nodes.

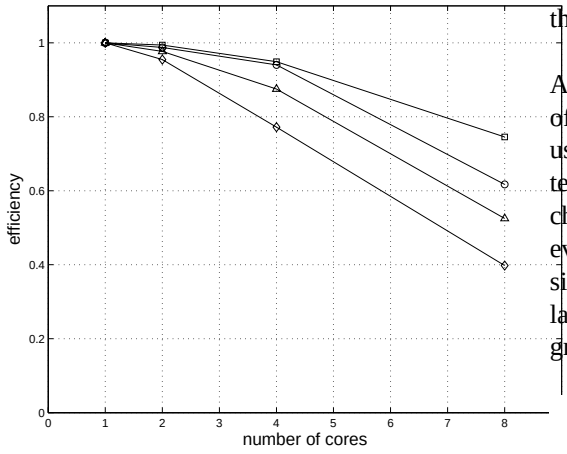


Figure 1. Algorithm 2 on a single node.

The test refers to the computation of (1) with $n=10$, $U=[0,1]^n$ and the following oscillating integrand function:

$$f = \cos\left(2\pi\beta_1 + \sum_{i=1}^{n} \alpha_i x_i\right)$$

where α and β are random values that determine the location and the sharpness of the oscillation. Therefore the reported results are the average on 10 executions obtained varying these parameters.

The figure shows the efficiency of the Algorithm 2 with different numbers of function evaluations, when the number of cores grows. The graph can be taken as a measure of the scalability of the Algorithm 2 on the cores of each node, i.e. as the ability to keep high efficiencies when the number of cores grows. Mainly for large problems (5217815 function evaluations), the Algorithm shows a very good scalability. However a different slope of the curves with 8 cores should be observed. This is because the CPUs have four cores each, so that the favorable ratio computation/access-to-data produces an execution with an extensive use of cached data up to four cores. On the other hand, the use of eight cores imply a more frequent access to the main memory charged to the second CPU.

Figure 2 reports the performance of Algorithm 2 measured with different numbers of cores with respect to the problem size, using the same computation from the previous test. As a measure of the performance we choose to use the number of function evaluations per second. Again we observe a significant gain in performance for sufficient large problems when the number of cores grows.

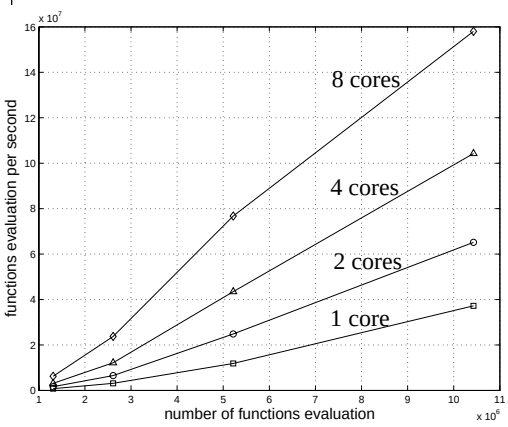


Figure 2. Performance of Algorithm 2.

Finally Figure 3 reports a scalability test on the 16 nodes of the system, using one core per node. This test is therefore aimed to measure the effectiveness of the upper level of the algorithm, i.e. the level responsible of the redistribution of the sub-domains among the

nodes. The distribution of the sub-domains with large error among connected nodes in a 2-dimensional periodical grid without global communication produce an high degree of scalability, that is the ability of the algorithm to keep high efficiency when the number of nodes grows.

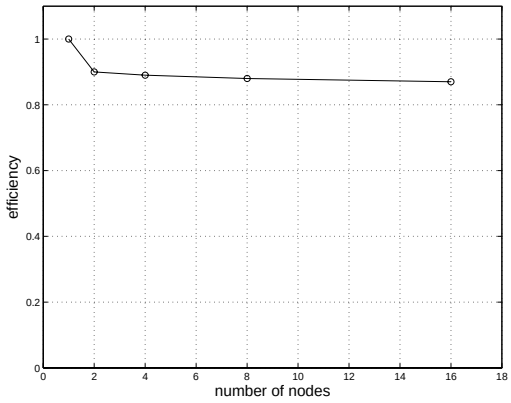


Figure 3. Scalability test.

ACKNOWLEDGMENTS

This work has been supported by Italian Ministry of Education, University and Research (MIUR) within the activities of the SCOPE project (PON Ricerca" 2000-2006 - Avviso 1575)

REFERENCES

[1] J. Berntsen, T. Espelid and A. Genz - Algorithm 698: DCUHRE - An adaptive multidimensional integration routine for a vector of integrals. - ACM Transaction on mathematical software, 17 (1991), pp. 452-456.

[2] R. Cools and P. Rabinowitz. - Monomial cubature rules since ``Stroud": a compilation. - Journal of Computational and Applied Mathematics, 48 (1993), pp.309-326

[3] M. D'Apuzzo and M. Lapegna. – Scalability and Load Balancing in Adaptive Algorithms for Multidimensional Integration – Parallel Computing vol. 23 (1997), pp. 1199-1210.

[4] A. D'Alessio and M. Lapegna - A Scalable Parallel Algorithm for the Adaptive Multidimensional Quadrature – in Parallel Processing for the Scientific Computing, R. Sincovec et al eds, SIAM 1993, pp. 933 - 936

[5] J. Dongarra, D. Gannon, G. Fox, and K. Kennedy - The Impact of Multicore on Computational Science Software**.** - CTWatch Quarterly, Volume 3 Number 1, February 2007

[6] D. Khaner and O. Rechard, - TWODQD: an adaptive routine for two-dimensional integration. - Journal of Computational and Applied Mathematics, 17 (1987), pp.215-234.

[7] A. Krommer and C. Ueberhuber – Computational Integration. – SIAM 1998

[8] G. Laccetti and M. Lapegna - PAMIHR. A Parallel FORTRAN Program for Multidimensional Quadrature on Distributed Memory Architectures. – in proceedings of EUROPAR99 Conference, LNCS 1685, Springer Verlag, pp. 1144 -1148.

Giuliano Laccetti is full professor of *Computer Science* in the Faculty of Sciences, University of Naples Federico II. He has been/is involved in several (national and european) Research Projects about Parallel and Distributed Computing, Scientific Computing, Grid Computing. Presently he is scientific coordinator of the University of Naples researchers group in the european EGEE III Project.

Marco Lapegna is associate professor in the Faculty of Sciences, University of Naples Federico II, where he teaches Parallel Computing, Operating Systems and Programming Laboratory. He has been involved in many national and European research projects about computing science topics, with special regard to parallel, grid and high performance computing.

Diego Romano is researcher at the Italian National Research Council where at present is on leave to study for his PhD on Computational and Computer Sciences at the University of Naples Federico II. His research interests are on parallel and distributed computing, focusing on topics such as Parallel Computer Graphics.

Valeria Mele is PhD student in Computational and Computer Sciences at the Department of Mathematics and Applications of the University of Naples Federico II.

Advances in Massive Model Visualization in the CYBERSAR project

M. Agus[1], F. Bettio[1], F. Marton[1], G. Pintore[1], and A. Zorcolo[1]
[1]*CRS4, Pula, Italy, vic@crs4.it*

***Abstract*—We provide a survey of the major results obtained within the CYBERSAR project in the area of massive data visualization. Despite the impressive improvements in graphics and computational hardware performance, interactive visualization of massive models still remains a challenging problem. To address this problem, we developed methods that exploit the programmability of latest generation graphics hardware, and combine coarse-grained multiresolution models, chunk-based data management with compression, incremental view-dependent level-of-detail selection, and visibility culling. The models that can be interactively rendered with our methods range from multi-gigabyte-sized datasets for general 3D meshes or scalar volumes, to terabyte-sized datasets in the restricted 2.5D case of digital terrain models. Such a performance enables novel ways of exploring massive datasets. In particular, we have demonstrated the capability of driving innovative light field displays able of giving multiple freely moving naked-eye viewers the illusion of seeing and manipulating massive 3D objects with continuous viewer-independent parallax.**

I. Introduction

OVER the last several decades, there have been significant advances in model acquisition, computer-aided design (CAD), and simulation technologies. These technologies have resulted in massive data sets of complex geometric models. These data include complex CAD environments, natural landscape models, scanned urban data, and various scientific simulation data. These massive data typically require giga byte size and even tera byte size. Handling such massive models presents important challenges to software and system developers. This is particularly true for highly interactive 3D applications, such as visual simulations and virtual environments, with their inherent focus on interactive, low-latency, and real-time processing. In the last decade, the graphics community has witnessed tremendous improvements in the performance and capabilities of computing and graphics hardware. Despite the continued increase in computing and graphics processing power, it is clear to the graphics community that one cannot just rely on hardware improvement to cope with arbitrarily large data sizes for the foreseeable future. This is not only because the increased computing power also allows users to produce more and more complex datasets, but also because memory bandwidth and data access speed grow at significantly slower rates than processing power and become the major bottlenecks when dealing with massive datasets. As a result, massive datasets cannot be interactively rendered by brute force methods. The real challenge is in designing rendering systems that capture as much of this performance growth as possible. When dealing with massive models, achieving high performance requires methods for carefully managing bandwidth requirements, controlling working set size, and ensuring coherent access patterns. Furthermore, given current multi-core CPUs and GPUs, solutions which can be formulated in a (data) parallel fashion will be able to continually take advantage of improved hardware performance. In this contribution, we provide a survey of the major results obtained within the project CYBERSAR in this field of research. Specifically, we will focus on the methods we developed to deal with huge models. The techniques exploit the

programmability of latest generation graphics hardware, and combine coarse-grained multiresolution models, chunk-based data management with compression, incremental view-dependent level-of-detail selection, and visibility culling. The models that can be interactively rendered with our methods range from multi-gigabyte-sized datasets for general 3D meshes or scalar volumes, to terabyte-sized datasets in the restricted 2.5D case of digital terrain models. Such a performance enables novel ways of exploring massive datasets. The rest of the paper is organized as follows: section II describes our multi-resolution out-of-core techniques for rapid high-quality visualization of textured digital terrain models and high resolution urban environments, section III deals with processing, distribution, and rendering of massive dense 3D meshes and huge complex 3D models, and section IV describes our method for rendering massive scalar volumes employing single pass GPU raycasting. Finally, section V describes our techniques for harnessing the power of novel 3D displays able of giving multiple freely moving naked-eye viewers the illusion of seeing and manipulating massive 3D objects with continuous viewer-independent parallax.

II. TERRAINS AND URBAN ENVIRONMENTS

Fig. 1. **Terrain rendering**: Interactive inspection of complex terrain datasets.

Real-time 3D exploration of remote digital elevation models built from high resolution imagery and elevation data has long been one of the most important components in a number of practical applications, and extensive research has been carried out in terms of methods and techniques for processing, distributing, and rendering very large datasets. The increased availability of broadband networks and high performance graphics PCs has made this technology, once limited to professional applications, increasingly popular, as testified by the success of Internet geo-viewing tools like Google Earth, NASA WorldWind, and Microsoft Virtual Earth. The efficient implementation of such tools requires a combination of technologies for adaptively rendering high quality terrain views at high frame rates and techniques for efficiently streaming data from the systems serving the terrain database to a large number of rendering clients. In this field, we introduced the BDAM methods [1], multi-resolution out-of-core techniques for rapid high-quality visualization of textured digital terrain models and high resolution urban environments. The BDAM framework introduced one of the first methods for rapidly generating seamless variable resolution surfaces by assembling precomputed patches. In the context of the CYBERSAR project, we improved the BDAM framework by introducing a compressed multiresolution representation and a client-server architecture for supporting interactive high quality remote visualization of very large textured planar and spherical terrains [2]. The approach incrementally updates a chunked level-of-detail BDAM hierarchy by using precomputed wavelet coefficient matrices [3] decoded from a compressed bitstream originating from a thin server. The structure combines the aggressive compression rates of wavelet-based image representations with the ability to ensure overall geometric continuity for variable resolution views of planar and spherical terrains with no need for run-time stitching [4]. The efficiency of the approach was demonstrated on a large scale interactive remote visualization of global and local terrains on ADSL networks. A library implementing this approach [5] has been incorporated into widely distributed geo-viewing systems [6], [7] (see figure 1). A survey on semi-regular multiresolution models for interactive terrain rendering has been also presented [8], illustrating the principal techniques based on tiled blocks and nested regular grids, quadtrees and triangle bin-trees triangulations,

as well as cluster based approaches.

III. Massive models

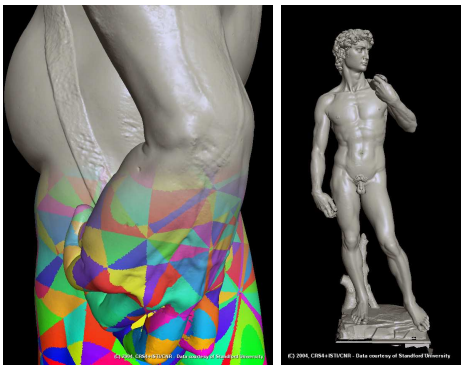


Fig. 2. **Massive models**: Digital Michelangelo David 1mm Rendering.

Many important applications domains, including 3D scanning, computer aided design, and numerical simulation, require the interactive inspection of huge geometric models. In this field, we considered methods for supporting inspection of surface models characterized by a high sample density, such as those generated by laser scanning. We also focused on methods able to support very large arbitrary surface models with high topological genus, highly variable depth complexity, fine geometric detail, "Bad" tessellations. Our main contributions are the introduction of Tetrapuzzles [9], a coarse grained multiresolution model based on hierarchical volumetric decomposition, that lead to the first GPU bound high quality technique for large scale meshes, and Far Voxels [10], an efficient approach for end-to-end out-of-core construction and interactive inspection that method tightly integrates visibility culling and out-of-core data management with a level-of-detail framework. The efficiency and generality of the approaches have been estensively demonstrated with the interactive rendering of extremely complex heterogeneous surface models on current commodity graphics platforms. The methods have been employed on a number of test cases, ranging from laser scans,including all Digital Michelangelo models, to isosurfaces, to extremely large CAD models, including the full Boeing 777 model (see figures 2 and 3). Fully interactive performance is obtained even for large windows on current single processor PCs.

In the context of the CYBERSAR project, we introduced a cluster parallel multiresolution renderer able to drive a lightfield display [11], [12]. The method is a parallel spatial 3D display-aware version of the Adaptive TetraPuzzles technique. It dynamically adapts model resolution by taking into account the particular spatial accuracy characteristics of the display.

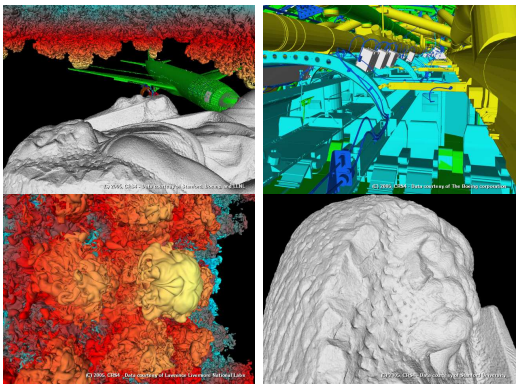


Fig. 3. **Massive models**: Interactive inspection of complex datasets.

Between the most relevant results, surveys and contributions to specialized courses have been presented in the most important computer graphics conferences [13], [14], [15], [16], illustrating the massive model rendering methods to the graphics community.

IV. Massive scalar volumes

The ability to interactively render rectilinear scalar volumes containing billions of samples on desktop PCs is of primary importance for a number of applications, which include medical visualization and numerical simulation results analysis. In this field, in the context of the CYBERSAR project, we introduced MOVR [17], an adaptive out-of-core technique for rendering massive scalar datasets within a single-pass

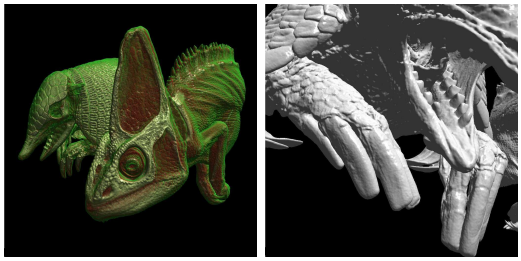


Fig. 4. **Massive scalar volumes rendering.** Interactive volume exploration of multi-giga voxels datasets on a desktop PC.

GPU raycasting framework. The method exploits an adaptive loader executing on the CPU for updating a working set of bricks maintained on GPU memory by asynchronously fetching data from an out-of-core volume octree representation. At each frame, a compact indexing structure, which spatially organizes the current working set into an octree hierarchy, is encoded in a small texture. This data structure is then exploited by an efficient raycasting algorithm, which computes the volume rendering integral by enumerating non-empty bricks in front-to-back order and adapting sampling density to brick resolution. In order to further optimize memory and bandwidth efficiency, the method also exploits feedback from the renderer to avoid refinement and data loading of occluded zones. The resulting method is extensible, fully adaptive, and able to interactively explore multi-giga-voxel datasets on a desktop PC (see figure 4).

V. Interactive visualization on novel 3D displays

Fig. 5. **Real-time interaction with the light field display.** Real-time inspection of a large volumetric model containing a CT dataset of a biological specimen. Pictures are taken with an hand-held camera at different viewing angles, in order to highlight the horizontal parallax of the light field display.

Fig. 6. **Massive model visualization on a 50Mpixel spatial 3D display.** The display has a screen dimension of 1.6x0.9 meters, and provides continuous horizontal parallax. Interactive rates in the manipulation of massive geometric models are guaranteed by a cluster-parallel spatial 3D display aware coarse grained multiresolution technique.

Multiscopic visualization is an emerging display technology that aims to reproduce three dimensional scenes by generating observer independent light fields. In the CYBERSAR project, we contributed to this field by developing methods for driving displays of this type [11], [18]. Specifically, we built a parallel multiresolution rendering system [11] driving a spatial 3D display able to give multiple viewers the illusion of seeing virtual models floating at fixed physical locations situated in a human–scale working volume (see figure 6). Each viewer sees the scene from their point of view and enjoys full, continuous, horizontal parallax without specialized viewing devices. The efficiency of the approach is demonstrated by an application supporting interactive manipulation of massive colored highly tessellated models on a large (1.6x0.9 meters) 50Mpixel display that allows for a room-size working space. We also developed a prototype medical data visualization system exploiting a small scale light field display and custom direct volume rendering techniques to enhance understanding of massive volumetric data, such as CT, MRI, and PET scans [18], [19], [20]. The system can be integrated with standard medical image archives and extends the capabilities of current radiology workstations. The system allows multiple untracked naked-eye users in a sufficiently large interaction area to coherently perceive rendered volumes as real

Evaluation tests demonstrated the usefulness of the generated depth cues and the improved performance in understanding complex spatial structures with respect to standard techniques.

REFERENCES

[1] P. Cignoni, F. Ganovelli, E. Gobbetti, F.Marton, F. Ponchio, and R. Scopigno, "BDAM: Batched dynamic adaptive meshes for high performance terrain visualization," *Computer Graphics Forum*, vol. 22, no. 3, pp. 505–514, Sept. 2003.

[2] F. Bettio, E. Gobbetti, F. Marton, and G. Pintore, "High-quality networked terrain rendering from compressed bitstreams," in *Proc. ACM Web3D International Symposium*, ACM Press. New York, NY, USA, April 2007, pp. 37–44. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Bettio:2007:HQN'

[3] E. Gobbetti, F. Marton, P. Cignoni, M. D. Benedetto, and F. Ganovelli, "C-BDAM – compressed batched dynamic adaptive meshes for terrain rendering," *Computer Graphics Forum*, vol. 25, no. 3, September 2006, proc. Eurographics 2006. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Gobbetti:2006:CCB'

[4] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, "Planet-sized batched dynamic adaptive meshes (P-BDAM)," in *IEEE Visualization*, 2003, pp. 147–154.

[5] "Ratman," Url: ratman.sourceforge.net, 2009.

[6] "Sardegna3d," Url: www.sardegna3d.it, 2007.

[7] "Italia3d," Url: www.italia3d.it, 2008.

[8] R. Pajarola and E. Gobbetti, "Survey on semi-regular multiresolution models for interactive terrain rendering," *The Visual Computer*, vol. 23, no. 8, pp. 583–605, 2007. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Pajarola:2007:SSR'

[9] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, "Adaptive tetrapuzzles: Efficient out-of-core construction and visualization of gigantic multiresolution polygonal models," *ACM Trans. on Graphics (SIGGRAPH 2004)*, vol. 23, no. 3, pp. 796–803, 2004.

[10] E. Gobbetti and F. Marton, "Far Voxels – a multiresolution framework for interactive rendering of huge complex 3d models on commodity graphics platforms," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 878–885, August 2005, proc. SIGGRAPH 2005. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Gobbetti:2005:FV'

[11] F. Bettio, E. Gobbetti, F. Marton, and G. Pintore, "Scalable rendering of massive triangle meshes on light field displays," *Computers & Graphics*, vol. 32, no. 1, pp. 55–64, February 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Bettio:2008:SRM'

[12] ——, "Multiresolution visualization of massive models on a large spatial 3d display," in *Proc. Eurographics Symposium on Parallel Graphics and Visualization*, Eurographics Association. Aire-la-Ville, Switzerland, May 2007. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Bettio:2007:MVM'

[13] A. Dietrich, E. Gobbetti, and S.-E. Yoon, "Massive-model rendering techniques: A tutorial," *IEEE Computer Graphics and Applications*, vol. 27, no. 6, pp. 20–34, nov/dec 2007. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Dietrich:2007:MMR'

[14] E. Gobbetti, D. Kasik, and S. eui Yoon, "Technical strategies for massive model visualization," in *Proc. ACM Solid and Physical Modeling Symposium*, ACM Press. New York, NY, USA, 2008, pp. 405–415. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Gobbetti:2008:TSM'

[15] S. eui Yoon, A. Dietrich, E. Gobbetti, D. Manocha, F. Marton, M. Agus, R. Pajarola, and P. Slusallek, "Interactive massive model rendering," SIGGRAPH Asia 2008 Course Notes, SIGGRAPH Asia 2008, Singapore, December 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Yoon:2008:IMM'

[16] D. Kasik, A. Dietrich, E. Gobbetti, D. Manocha, F. Marton, P. Slusallek, A. Stephens, and S. eui Yoon, "Massive model visualization techniques," SIGGRAPH 2008 Course Notes, SIGGRAPH 2008, San Diego, August 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Kasik:2008:MMV'

[17] E. Gobbetti, F. Marton, and J. Iglesias Guitián, "A single-pass GPU ray casting framework for interactive out-of-core rendering of massive volumetric datasets," *The Visual Computer*, vol. 24, no. 7-9, pp. 797–806, 2008, proc. CGI 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Gobbetti:2008:SGR'

[18] M. Agus, F. Bettio, A. Giachetti, E. Gobbetti, J. A. I. Guitián, F. Marton, J. Nilsson, and G. Pintore, "An interactive 3d medical visualization system based on a light field display," *The Visual Computer*, vol. 25, 2009, to appear. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2009:'

[19] M. Agus, A. Giachetti, E. Gobbetti, J. Iglesias Guitián, and F. Marton, "Towards advanced volumetric display of the human musculoskeletal system," in *Eurographics Italian Chapter Conference*. Conference held in Salerno, Italy: Eurographics Association, June 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2008:AVD'

[20] M. Agus, E. Gobbetti, J. Iglesias Guitián, F. Marton, and G. Pintore, "Gpu accelerated direct volume rendering on an interactive light field display," *Computer Graphics Forum*, vol. 27, no. 3, pp. 231–240, 2008, proc. Eurographics 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2008:GAD'

Grid@Trieste: A Metropolitan Area Grid

C. Vuerli[1,2], R. Pugliese[3], F. Liello[4] and F. Pasian[1,2]
[1]*INAF - Information System Unit, Trieste, Italy*
[2]*INAF - Osservatorio Astronomico di Trieste, Trieste, Italy*
[3]*Sincrotrone Trieste SCpA, Trieste, Italy*
[4]*University of Trieste, Trieste, Italy*

Abstract — **The new MAN (Metropolitan Area Network) in Trieste is in place since June 2008. All scientific Institutes composing the wide and rich metropolitan scientific park met and established an ATS (Associazione Temporanea di Scopo) to design and start up the new MAN based on Fiber Optic Rings able to bring up to 32 wavelengths (λ) and each of them able to transport 1 Gbps of information. The architecture has been thought to guarantee a full connectivity even in case of failure of one or more physical connections. Each Institute owns one of more λs each of them dedicated to a specific purpose (connections to the GARR POP, intra-Lightnet connections, intra-Institute connections, specific multi-Institution services or applications). Until spring 2009 Lightnet will be in a transient phase where the whole set of λs is not activated yet. Connections are granted through VLANs each of them with a nominal bandwidth of 100 Mbps.**
One of the 1 Gb λs of Lightnet has been reserved to Grid@Trieste, an experimental project set up in 2005 whose target was the creation of a stable metropolitan Grid Infrastructure for the benefit of all scientific Institutions in Trieste. The project was created and managed by Sincrotrone Trieste, ICTP, SISSA, OGS, the University, and INAF. The experimental phase of the project was very useful to set up all Grid tools and services necessary to make the infrastructure self-consistent and to port on this infrastructure a number of scientific applications contributed by the various scientific communities. The most important obstacle for a further expansion of the infrastructure was represented by the limited-bandwidth for inter-Institute connections. Thanks to the advent of Lightnet this obstacle is now definitely removed and further expansions of Grid@Trieste become now possible, both from the geographical point of view going beyond the metropolitan boundaries and of new communities (also non scientific) that could join the project.

Index Terms — **Metropolitan Area Network, Lightnet, Metropolitan Area Grid, Applications, Grid Tools.**

I. INTRODUCTION

THIS paper illustrates the project aimed at creating a "Metropolitan Area Grid" in Trieste that involves all major scientific Institutions and whose goal is the creation of a Grid infrastructure that participating Institutes can exploit to run their applications and carry out their scientific tasks.
The necessary precondition to guarantee the success of the Metropolitan Grid Infrastructure is an efficient, broadband Metropolitan Area Network connecting all participating Institutions. This precondition has been fulfilled by the advent of Lightnet, a Metropolitan Area Network based on Optical Fiber Rings and able to bring up to 32 wavelengths (λ) each of them able to transport 1 Gbps of information.

A. Lightnet

Lightnet, the project to set up a high throughput metropolitan area network in Trieste dates back 2005. It is a Fiber Optic network that connects the most important scientific Institutions in Trieste. In order to set up Lightnet an agreement has been established among all participating Institutions. For this purpose they created an ATS (Associazione Temporanea di Scopo) led by the University of Trieste. A total budget of 2 M€ has been allocated for the Lightnet project; a fraction of this budget (1.5 M€) has been used to deploy 250 km of optical fibers that constitute the

physical infrastructure of the MAN; the remaining fraction (500 K€) will be used to procure the necessary network devices necessary to connect the various nodes and make operative the optical rings.

Twelve scientific institutions of Trieste participate to the project, each of them contributing with a fraction of the necessary funds. Among the scientific institutes that support the project we find ICTP (the International Centre of Theoretical Physics), the Area Science Park, the Sincrotrone Trieste SCpA, SISSA (Scuola Internazionale Superiore di Studi Avanzati), GARR (Gruppo Armonizzazione Reti per la Ricerca) being it the body that maintains and operates the whole network for the scientific research in Italy, ISMAR (Istituto di Scienze Marine), OGS (Osservatorio Geofisico Sperimentale), INAF (National Institute of Astrophysics), the "Globo" association and the School of Music "Tartini".

1) Physical Structure

We go more in deep here in describing technically the optical fiber network. The figure below gives the layout of the MAN, as it is at the time of preparation of the present paper.

The figure well highlights the optical ring based structure of the network.

The physical transport layer consists of one or more pairs of mono-modal optical fibers. All Institutions participating to the project are directly connected to the MAN. Wherever possible, the physical paths of the fiber are chosen in order to create a topological structure of interconnected rings; this structure, thanks to the fact that paths are doubled, ensures the necessary redundancy.

2) Logical Structure

The physical structure described above makes possible a high level of flexibility for what concerns the connections between two different points of the network. The definition of the various logical connectivity paths, therefore, can be done by taking into account the requirements of the single Institutes.

A GARR Giga POP has been also set up and located at CSIA, the Informative Services Centre of the University, as agreed with GARR itself.

Logically separated connections, each of them with a bandwidth capacity not less than 1 Gbps have been dedicated to:

- Connect the headquarters of each Institution with the GARR POP.
- Create private connections (intra-Institution) for all those Institutes with two or more branches geographically distributed.
- Interconnect the headquarters of all participating Institutions (local connections).
- Create logical connections for special purposes, usually related to a specific protocol or project. A logical connection of 1 Gbps, for instance, has been reserved to the Grid@Trieste project.

Logical paths are built thanks to the wavelength separation of type "Coarse" (CWDM) or "Dense" (DWDM) in conjunction with the VLAN Gigabit Ethernet technology.

This allows a full separation of the network traffic at the lowest possible level as permitted by current available technologies and makes

possible an adequate level of privacy (optical VPN). In this way it is also possible a sharing of the fiber, so that its capacity is virtually enhanced and its management is delegated to each single Institution.

For what concerns the utilization of the optical paths created on the network, two options are possible:

- Primary connection of the headquarter of each institution to the GARR Giga-POP. Eight 1 Gbps lambdas (λ) and one 1 Gbps VLAN have been allocated for this purpose.
- Local public connectivity (inter-Institutions) or private connectivity (intra-Institution). Currently one single λ has been reserved for local inter-Institutions connections and additional six λ have been allocated for the intra-connections for all Institutions that need this capability.

3) Current status and future evolution

During the summer 2008 Lightnet entered in its operational phase and in a couple of months all participating Institutions joined the new network infrastructure. This was the first step in the process of activation of Lightnet. In this transient phase only a fraction of Lightnet is enabled and the various connections previously described are mapped over 100 Mbps VLANs. It is foreseen that Lightnet will enter its stable full-operative phase in the second half of 2009. At the time of preparation of this paper the ATS is providing the necessary formal steps to acquire the optical network devices requested to fully enable the optical network infrastructure. At the same time each Institute will procure the border router able to serve all its connections (λs).

Because Lightnet requires staff personnel to manage and operate the optical network it is necessary to create an autonomous and permanent legal body able to handle its own funds and to regulate all relationships among the participating Institutes. For this reason the ATS established to start up Lightnet will evolve in a more stable legal entity during 2009.

B. Grid@Trieste

Grid@Trieste is the initiative aimed at setting up a stable Metropolitan Grid Infrastructure to allow participating Institutes to operate with it for the benefit of their scientific communities. As for any other Grid infrastructure, the biggest added value brought by Grid@Trieste is to foster the scientific and technological collaborations among all Institutions located in the metropolitan area of Trieste, extremely rich for what concerns the scientific and technological research centers.

The sharing of resources (of various types) is the key concept for this collaboration. As any other Grid infrastructure, Grid@Trieste allows the sharing of its own resources and the exploitation of resources owned by other partners of the project.

The relevance of an initiative like Grid@Trieste is remarkable given the high number of potentially involved Institutions and research groups.

1) The experimental phase

Grid@Trieste started as an experimental project in February 2005 thanks to the initiative of six big research Institutes in Trieste, namely ICTP, INAF, OGS (Osservatorio Geofisico Sperimentale), Sincrotrone Trieste SCpA, SISSA/Democritos and University of Trieste.

On February 8th 2005 the founding participants met at INAF Trieste in the framework of a workshop in which all Institutes presented their past or in progress Grid-related activities and exchanged their own expertise and know-how in relation with this technology. The meeting was attended also by potential industry partners. The founders of the experimental project, in fact, realized that synergies with the commercial world and industry should have been investigated right away given that the potentiality of the Grid can have a great impact also on commerce and

industry.

Each participant brought in this meeting its own experience coming from various projects. It is worth to mention here the main projects from which participants took ideas and software tools during the initial setup of Grid@Trieste.

a) The EGRID project [1] (SISSA/Democritos) aimed at implementing a national facility for processing economic and financial data using computational grid technology; as such, it acted as the underlying fabric on top of which partner projects, more strictly focused on research in itself, developed end-user applications.

b) The GridCC project [2] (Sincrotrone Trieste SCpA) funded by the European Community, aimed at providing access to and control of distributed complex instrumentation; the goal of GridCC was to exploit grid opportunities for the secure and collaborative work of distributed teams, in order to remotely operate and monitor scientific equipment using the Grid's massive memory and computing resources for storing and processing data generated by this kind of equipment.

c) Grid.it [3] (INAF), an Italian project with a strong interdisciplinary character aimed at defining, implementing and applying innovative solutions for network computing enabling platforms, oriented towards scalable VOs and based on the "Grid Computing" paradigm. The research topics in the project spanned from high performance photonic networks, innovative middleware services, high performance programming environments and so forth.

At the end of the workshop all participants fully agreed on the opportunity to start with the experimental project.

During the one-year experimental phase of the project frequent meetings took place with the participation of all partners. Restricted technical working groups were also formed, each of them focused on a specific technical aspect of the project.

The main technical areas included:

- The middleware to use for Grid@Trieste.
- Specific tools and services (both for Grid site maintainers as well as for application developers) that participants considered of strategic importance for the success of the experimental project.
- The authentication/authorization mechanisms to allow users to get certificates and start using the Grid@Trieste infrastructure.
- The identification of a pool of pilot applications representative of all scientific disciplines involved in the project used to disseminate the project results within the various scientific communities and attract in this way new potential users and research groups.

The so formed technical working groups worked hardly during the experimental phase of the project.

For what concerns the middleware, during the experimental phase the LCG-2 [4] middleware was adopted. During this phase the staff of the EGID project also set up and maintained the services of Grid@Trieste. In particular it was set up a BDII, a Resource Broker, a LFC catalogue and the VOMS server for the Virtual Organization explicitly created for the project.

Because a CA was previously created for the EGRD project, the same CA was used to authenticate/authorize users in Grid@Trieste by issuing proper user and host certificates.

A live CD was also prepared for Grid@Trieste. This important tool, introduced the first time for the EGRID project, was based on LCG-2/EDG and on Knoppix system; it made possible to test a Grid node and access Grid@Trieste without the need of installing the software residing on the live CD. The live CD was distributed with a simple configuration interface that allowed an easier configuration of the Grid node with its typical services.

The work dedicated to user applications and

to dissemination aspects was the most challenging. As for many other Grid infrastructures, also in Grid@Trieste the objective of attracting new users proved to be not a trivial task. The best way to do that is to identify a pool of pilot applications in order to cover the whole spectrum of scientific domains involved in the experimentation. Within each participating Institute single users and research groups were contacted and invited to submit their applications to Grid@Trieste. At the end of the experimental phase the pool of so identified applications were reported in the closing workshop.

A Grid@Trieste open day (that lasted indeed for a whole week end) was also organized. The purpose of the initiative was to completely open the Grid infrastructure of the project and invite all users to freely test the metropolitan Grid by submitting their jobs. In view of this event all Institutes shared their own available resources so that it was possible to carry out tests with a significant amount of them.

The experimental phase of the project clearly demonstrated that the success of a Metropolitan Area Grid is possible only if a broadband and efficient underlying physical network infrastructure is in place. People participating to the Grid@Trieste experiment were fully aware of this and for this reason they monitored very closely the progress of Lightnet and kept constantly in touch with Prof. F. Liello, promoter and P.I. of Lightnet. During this phase it was agreed to reserve one of the available 1 Gbps λs of Lightnet to Grid@Trieste (to its services). This dedicated λ will "touch" all Institutes being part of the metropolitan Grid. In this way the efficiency issues that emerged during the experimental phase will be completely overcome.

At the end of January 2006 the experimental phase of Grid@Trieste, carried out by participating Institutes with no funds available, terminated. To officially close this first phase a workshop was organized at the Area Science Park; Members of the Regional Government having the responsibility over the scientific and technological research and innovation were invited to attend the event as well as the Heads of all participating Institutions. After the welcome messages addressed by the Authorities, the status of some key strategic projects for what concerns the Grid were illustrated (Grid@Trieste, Lightnet, GridCC); the pilot applications belonging to several scientific domains (computational fluido-dynamics, astrophysics, new massive computing opportunities for geophysics) and chosen as demonstrators were illustrated as well together with relevant Grid-related research activity carried out at Sincrotrone Trieste (portals, several Grid-enabled tools, simulations). The workshop closed with a round table aimed at discussing future developments and perspectives of a regional Grid infrastructure, with the participation of people representing the Regional Government, local scientific Institutions and the Industry.

2) Current Status

After the conclusion of the experimental phase, the project entered in a standby state. All activities were frozen due to lack of funds and despite the potentiality of the project that clearly emerged from the experimental phase, it was not possible to go ahead with planned activities.

Fortunately the metropolitan Grid did not completely die and this thanks to the activities carried out at Sincrotrone Trieste, where the local group operating on the various Grid-related projects, mainly GridCC until the end of August 2007 and now DORII [5], a project supported by the European Commission within FP7, continued to maintain the infrastructure of Grid@Trieste. Thanks to this activity it was possible to keep the infrastructure in line with new releases of the LCG and gLite middleware. People at Sincrotrone Trieste took also care of completely reinstall the main Grid services and equipped the infrastructure with a minimum pool of computing and storage

resources. Some tools developed in the framework of GridCC were also installed, so apart from the classical CE and SE, the Metropolitan Grid is now equipped with the VCR (Virtual Control Room), a powerful set of collaboration tools developed by Sincrotrone Trieste that allows users to build complex workflows, which are then submitted to the ES (Execution Services), and to directly monitor and control remote instruments in real-time. In the future another tool developed in the framework of GridCC, the IE (Instrument Element), could be also installed allowing research groups dealing with instruments and sensors of various nature to operate with them via the Grid.

3) Future Perspectives

The advent of Lightnet definitely opens new interesting perspectives for the Metropolitan Area Grid. Lightnet will enter in its fully operative production phase in late summer/early autumn 2009 and at that time the available bandwidth and the sophisticated technology on top of which Lightnet is built will make possible local scientific and technological collaborations before unimaginable. The Grid is one of the technologies that can take great advantage from this scenario. An underlying highly efficient physical network infrastructure means a high level of efficiency of Grid services and this is a crucial aspect in the effort of capturing new applications and research groups, the only reason that can justify financial investments in this technology.

Lightnet moreover could later expand beyond the Trieste metropolitan area and become a Regional Grid infrastructure or even an inter-Regional European Grid by involving the neighbouring regions of Austria, Slovenia and Croatia. As already explained above the future of the Metropolitan Area Grid is, for technological reasons, closely correlated to Lightnet, so the expansion of Lightnet outside the national boundaries could be followed by an analogous expansion of the regional Grid infrastructure with new open perspectives from the point of view of wider scientific collaborations and also new opportunities to get funds by the European Union.

For the time being, however, it is necessary to quickly resume the project for a Metropolitan Area Grid; this task cannot be entirely done in an unfunded way. Institutes operating in the Trieste area have to meet and sketch out a planning for a re-launch of the project. It is necessary to work together on a concrete proposal to get the necessary funds. A good starting point, however, is the re-establishment of the infrastructure of Grid@Trieste in terms of Grid services and tools and by sharing at least part of the resources owned by each Institute. If possible, this process should be completed in time for a full exploitation of the 1 Gbps λ that Lightnet will allocate to Grid@Trieste.

II. CONCLUSION

The experimental initial phase of Grid@Trieste proved that a Metropolitan Area Grid is a great scientific and technological opportunity for Trieste where a remarkable number of scientific Institutes is concentrated. The advent of Lightnet, moreover, provides the underlying physical network necessary for Grid services to run efficiently. In this way an important technological gap is definitely overcome. A full re-launch of Grid@Trieste, nevertheless, requires a renewed commitment by those Institutes that participated to the early phase of the project now in a standby state. The list of short-run goals to pursue should be:

a) re-establish the Grid@Trieste infrastructure through a restoration of the basic Grid services and an adequate amount of resources contributed by all Institutes that accept to participate to the re-launch of the project;

b) investigate new opportunities to get funds for the project at local, national and European

level. Once the project has been resumed and its full operability as a metropolitan Grid in Trieste is achieved, geographical expansions could be considered; in this way wider user communities might benefit of the Grid with new opportunities for scientific and technological collaborations.

ACKNOWLEDGMENTS

Authors thank all colleagues that in these last years operated to set up and manage Lightnet in a fast, efficient and reliable way, and colleagues who participated to the experimental phase of Grid@Trieste.

REFERENCES

[1] http://www.egrid.it/
[2] http://www.gridcc.org/cms/
[3] http://grid.it/
[4] http://lcg.web.cern.ch/LCG/
[5] http://www.dorii.eu/?DokuWiki=aa4331648d5a985c915fadcb36f1e0bb

C. Vuerli is Associate Astronomer. He took his degree in Computer Science in 1985. Currently he's involved and coordinates all Grid-related projects at INAF-SI and INAF-Astronomical Observatory of Trieste. He is responsible for managing activities finalized to make the Grid a suitable tool for astrophysical applications. Since August 2005 he is staff of SI, the INAF structure having in charge the national coordination of all services related with data processing, data archiving and network infrastructure. Since January 2007 he coordinates the Astrophysical cluster (A&A cluster) created within the EGEE project.

R. Pugliese is a Research Coordinator at Sincrotrone Trieste S.C.p.A. where he is leading the Scientific Computing Group. Since October 2002 he is also Professor of E-Commerce at the University of Udine. His research interests include Web Based Virtual Collaborations and Grid technologies. Roberto Pugliese was the technical coordinator of the GRIDCC project and is currently coordinating the Applications workpackage of the DORII project.

F. Liello took his degree in Physics at the University of Rome in December 1973. He carried out his research activity in the fields of molecular biology (University of Rome), of elementary particles on several accelerators (Rome, CERN, Trieste), of cosmic rays physics both through ground-based and satellite-based experiments; nowadays he's still active in these research fields. He's one of the founders of the Italian data transmission network for academy and research (GARR) and of the international counterpart networks (TEN-34, TEN-155, GEANT). He was President of the European Consortium for the Research Networks and President of the Commission for Computing and Networks of INFN. He's currently member of the Technical-Scientific Committee of GARR and coordinates international projects aimed at enabling remote access to ESO and INFN centres of excellence located in Latin America.

F. Pasian is Full Astronomer. He's the Director of INAF-SI (INAF-Information Systems) for the national coordination of all services related with data processing, data archiving and network infrastructure used in astrophysics. He has been active in the field of astronomical image processing, data archiving and data processing. He has led many projects both at national and international level such as the archive of the Hubble Space Telescope (HST), the archive at the Galileo National Telescope (TNG), the Data Processing Centre of the Low Frequency Instrument of the Planck mission of ESA. He's author/co-author of about 300 publications and technical notes.

Integration of Matlab® with the COMETA Infrastructure: Case Study with a Scientific Application

G. Castiglia[1], M. Cipolla[2,4], P.P. Corso[1,4] and D. La Porta[3,4]

[1]Dipartimento di Scienze Fisiche ed Astronomiche,
Università degli Studi di Palermo, via Archirafi 36, 90123 Palermo, Italy and CNISM.
[2]Dipartimento di Matematica ed Applicazioni
Università degli Studi di Palermo, via Archirafi 34, 90123 Palermo, Italy
[3]Dipartimento di Ingegneria Informatica
Università degli Studi di Palermo, Viale delle Scienze - Edificio 6, 90128 Palermo, Italy
[4]Consorzio COMETA – Via Santa Sofia 64 – I-95123 Catania, Italy

***Abstract*— We have integrated the MATLAB Distributed Computing Server™ (MDCS) with the gLite middleware of the Grid infrastructure of the COMETA Consortium. Our work has been focused on the identification of the critical MATLAB functions used to set up parallel, distributed or simple programs and also on the typical environment settings involved in the integration. Our contribution is just intended to better clarify how the integration has been performed, both in terms of general principles and approach.**

Furthermore, we have tested the performances of MDCS in the COMETA parallel environment, by means of two specific case studies.

***Index Terms* — COMETA, gLite middleware, MATLAB Distributed Computing Server, Grid.**

I. INTRODUCTION

FROM the beginning of 80's the computational power available for the scientific community has evolved from single workstation to network of super computers. Many current scientific problems require resources that common workstations do not provide both in terms of computational power and in terms of data storage. In this context, ICT technologies provide solutions that allow to arrange computational facilities at different scale networks, from small local networks of workstations to Virtual Organizations (VOs) and Grid systems.

In order to take advantage of these resources, new solutions have to take into account parallelism in their implementation. The design of a parallel algorithm is a complex task since parallel solutions are strongly dependent on the nature of the problem and require the knowledge of many concepts concerning programming and low-level machine architecture.

The scientific community considers the MATLAB software a very useful tool since it allows to develop algorithms for a vast variety of problems without taking into account low-level issues usually arising in programming. Since roughly five years, MATLAB provides a complete set of tools for designing parallel algorithms whose performances have been investigated in [1].

In this paper we introduce the reader to a more detailed overview of the integration of MATLAB with a Grid system, namely the COMETA Grid. The MATLAB Parallel Computing Toolbox™ allows the user to develop and run parallel applications starting working on a single workstation, the process of submission of a parallel program (or simply job) being managed transparently by the MATLAB itself. Other information regarding a similar integration can be found in [2].

The effective extensive MATLAB parallel computing environment is represented by the MATLAB Distributed Computing Server™ which, on the COMETA Grid consists of a 128 core license. Since our main issue concerns the interface between the MATLAB client and the

Grid, the following Sections focus on this problem and give some detail on the adopted solutions. In order to evaluate the performances on our system we investigate two case studies.

Section II gives a brief explanation about the organization of the principal components of a Grid environment. Section III explains how MATLAB can be interfaced to a Grid system and run processes. Section IV describes the physical models considered for the benchmark and, finally, Section V shows the results.

II. Overview of the Grid Environment

The integration of a generic software with a Grid infrastructure is usually achieved by configuring both the client side (the generic software) and the host side (the Grid infrastructure). The main components involved in our specific integration are: the MATLAB client, the User Interface (UI), the Logical File Catalog (LFC), the Computing Element (CE), the gLite middleware (for the COMETA infrastructure) and the Worker Nodes (WN).

Grid users access the resources on the Grid by means of the UI, where all data needed for the job submission are stored.

The job submission is managed by the gLite middleware. The interface with gLite is achieved by means of jdl scripts, which allow to specify all the requirements for a specific job. The main information we usually need to supply with jdl scripts are:

- the input file names;
- the output file names;
- the number of processors;
- the job queue.

Once a jdl script is executed, gLite checks if the resources required by the job are available on the grid. The information relative to the Grid resources are stored and published by the CE; gLite thus checks if the requirement specified by the jdl script matches the available resources. The job input data (if any) is copied in a memory location which is shared by the workers. After the computation, all the output data are stored by a specific WN, namely, the master worker node; it represents the worker node that produces the required output.

For large data set (greater than 100 MB) it necessary to manually copy the data in some location where they will be retrieved by the workers; for this purpose it is necessary to use the LFC catalog. The copy of the data to and from the catalog is managed by a user-provided shell script file, specified in the jdl and executed on the master node. Finally the output is retrieved either from the master node or from the catalog and copied back into the UI.

III. MATLAB on the COMETA Grid

In order to be able to submit a MATLAB job to the COMETA Grid, it is necessary to interface MATLAB with the gLite middleware. It is thus necessary to use the UI as a link between the MATLAB client and gLite. The problem can be solved by simply setting an ssh password-less connection between the client and the UI; MATLAB can thus copy data and execute commands on the UI.

Once connected to the UI, it is necessary to submit the job. MATLAB provides a set of functions for submitting jobs by using a generic scheduler interface. These functions need to be customized according to the environment configuration. The most important elements of the scheduler interface to be modified are the submit function and the decode function. The first is used to set up the environment, the latter is executed by the WNs to retrieve all the information needed to run jobs. The main operation performed by the submit function are:

- generation of the jdl script on the local machine and copy of this file to the UI;
- copy of the job data to either the UI or the Catalog;
- execution of remote commands on the UI.

The jdl script itself requires the following scripts to be used during the job submission: pre.sh, gLiteWorker.sh and lsfParallelWrapper.sh.

Once the jdl is executed, it runs the pre.sh

script which is necessary to set up all the environment variables needed by the script gLiteWorker.sh. These environment variables mainly specify the decode function each WN is going to execute, the storage location, the MATLAB executable location, the input arguments, the number of requested cores, etc.

After having set all the necessary environment variables, the gLiteWorker.sh script copies all the files onto a shared storage which is used by the workers to retrieve the input data. The gLiteWorker.sh script is thus ready to execute the lsfParallelWrapper.sh script which starts all the processes needed for the computation. When the WNs start their execution, the decode function is used to locate the input data by using the environment variables previously set by the gLiteWorker.sh script. At the end of the computation, the gLiteWorker.sh script collects the output data in the master WN; such data is then available and can be retrieved.

In our integration we have solved the problem of the data dimension by introducing a check into the submit function and into the gLiteWorker.sh.

IV. The Case Studies

In order to test the performances of the MDCS, we have solved numerically the equations concerning two typical physical systems:

- the heat diffusion on a 2D plate;
- the temporal evolution of a 1D Hydrogen molecule driven by a laser field.

The 2D heat equation, describing the diffusion of heat on a rectangular plate, is given by:

$$D\frac{\partial}{\partial t}u = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)u \qquad (1)$$

where $u \equiv u(x, y, t)$ is the time-dependent heat distribution on the plate and *D* the diffusion constant.

By using the *finite-difference equation approximation* [2]-[4], eq. (1) becomes:

$$u_{i,j}^{k+1} = A\left[\, u_{i-1,j}^{k} + u_{i+1,j}^{k} \,\right] + \\ + B\left[\, u_{i,j-1}^{k} + u_{i,j+1}^{k} \,\right] + \left(1 - C\right)u_{i,j}^{k} \qquad (2)$$

where $A = \dfrac{\Delta t}{D(\Delta x)^2}$, $B = \dfrac{\Delta t}{D(\Delta y)^2}$ and $C = 2\left(A + B\right)$. Δx and Δy are the spatial step grid in both *x* and *y* direction, Δt is the temporal step and $u_{i,j}^{k} \equiv u(x_i, y_j, t_k)$, namely the temperature at the grid point (x_i, y_j) at time t_k (*i*, *j*, *k* integers).

Eq. (2) implies that the temperature at a given grid point depends on the value of the temperature at the nearest points at the previous time step. Fig. 1 shows a graphical representation of such dependence.

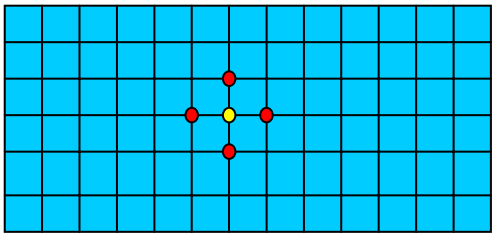


Fig. 1: Graphical representation of eq. (2).

If we want to solve equ. (2) by a parallel numerical approach, the graphical representation in Fig. 2 shows that each processor must communicates the boundary data to the near processor. Actually, for example, it is evident that processor 3 (P3), in order to evaluate the temperature in the yellow point, needs to know both the value of temperature at the red points (values that it knows) and at the green point (value that only P2 knows).

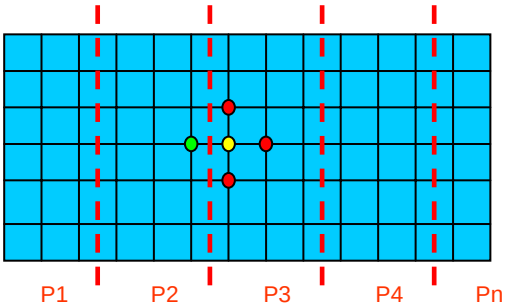


Fig. 2: Graphical representation of eq. (2) in the case of a parallel numerical approach (P*i* indicates the *i*-th processor).

Regarding the second model, we have considered a one-dimensional model, with fixed nuclei, of the Hydrogen molecule driven by a laser field. The equation governing the time evolution of such system is the Schroedinger equation (expressed in atomic unit):

$$i\frac{\partial}{\partial t}\psi = \left[-\frac{1}{2}\left(\frac{\partial^2}{\partial x_1^2}+\frac{\partial^2}{\partial x_2^2}\right)+U(x_1,x_2,t)\right]\psi \qquad (3)$$

where x_1 and x_2 are the positions of the two electrons, $U(x_1,x_2,t)$ the term containing both the electrostatic energy and the interaction energy with the field and $\psi \equiv \psi(x_1,x_2,t)$ the wavefunction of the system.

To solve this equation, we have developed a MATLAB parallel code based on the technique of the *split-operator* method.

The main difference with the previous code consists on a massive communication between the processors. Actually the *split-operator* method is characterized by the fact that some calculations are evaluated in the momentum space. This means that it is necessary to perform *bi*-dimensional Fourier transforms that we have evaluated by using the `FFT` (*fast-fourier transform*) MATLAB function. Unfortunately, on a parallel environment, the data are distributed among the processors in a way similar to what shown in Fig. 2. In order to perform the Fourier transform along the rows, it is thus essential that each processor communicates its own data to the others processors. To evaluate the `FFT` in the row direction, it is necessary to rearrange the data by means of a transposition.

By this description, it is evident the main difference between these two models: in the first case, each processor communicates only with the adjacent processors and the exchanged data are of the order of N (N being the number of grid points in each direction assuming a square numerical grid); on the contrary, in the second case, each processor communicates with all the other processors and the order of exchanged data scales as N^2. Furthermore, in the latter case the requested communications for each step, when are used N_p processors, scale as N_p^2, more expensive in comparison to the first model where the requested communications scales as N_p.

In the next Section we will show that the communications between processors play an important role on the performances of the code.

V. Benchmark

In Fig. 3 we show the performances of the MDCS on the COMETA parallel environment when solving the heat equation code.

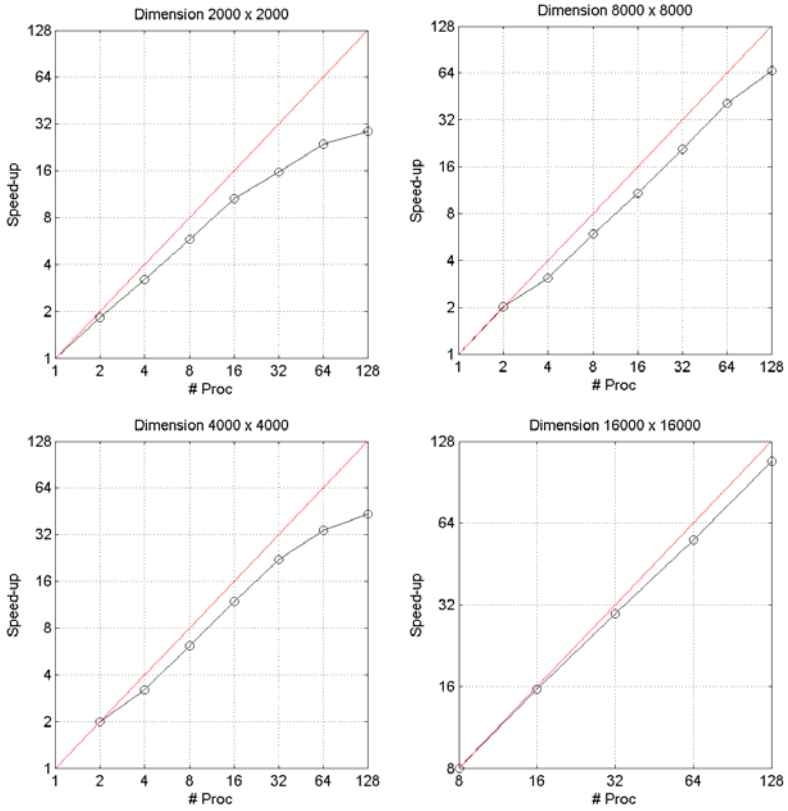


Fig. 3: *Speed-up* (black line), for different matrix dimensions, calculated in the case of the heat equation code. The red line shows the theoretical expected *speed-up*. The axis scale is $\log_2$-$\log_2$.

As it is evident, the *speed-up* is better for larger box. This is due to the fact that, for larger box, the time spent for the communications is negligible compared to the time taken to perform the calculations.

In Fig. 4 we show analogous results obtained solving the one-dimensional Schroedinger equation for the Hydrogen molecule. In the left subplot, showing the results for a matrix dimension of 4096x4096 points, it is evident a good scaling of the speed-up, up to 16 processors, followed by an abrupt decrease of performances. The trend appears to be better in the case of larger matrix (8192x8192 points).

If we compare the last results with those shown in Fig. 3, it is clear that the performances are worse in the case of the Schroedinger equation code. The reasons for this behavior are mainly two: the order of exchanged data (N^2) which mainly deals with the communication bandwidth and the number of activated communications between the processors (of order N_p^2) mainly related to the latency of communications.

VI. Conclusion

In this paper it is shown the integration of the MATLAB Distributed Computing Server with the Grid infrastructure of the Cometa Consortium. In particular, they are pointed out both the main logical steps to follow and the critical script files to customize in order to make possible such an integration.

By means of two different case studies, it is shown that better performances are achieved

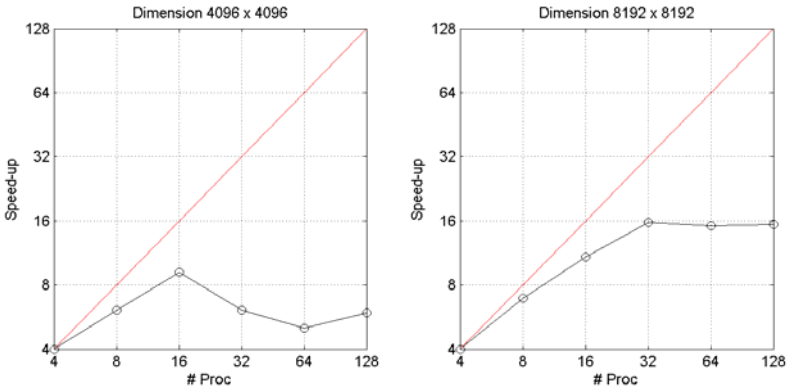


Fig. 4: *Speed-up* (black line), for different matrix dimensions, calculated in the case of the Hydrogen molecule code. The red line shows the theoretical expected *speed-up*. The axis scale is $\log_2$-$\log_2$.

for large numerical grids. On the contrary, the communication overhead is relevant with respect to the overall execution time in the case of small matrices.

ACKNOWLEDGMENTS

The authors would like to thank the The MathWorks® development team.

This work has been partially financially supported by the University of Palermo Research Program and by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione"\ (PON 2000-2006). More information is available at *http://www.pi2s2.it* and *http://www.consorzio-cometa.it*.

REFERENCES

A. J. Chakravarti, S. Grad-Freilich, E. Laure, M. Jouvin, G. Philippon, C. Loomis, E. Floros , “Enhancing e-Infrastructures with Advanced Technical Computing : Parallel MATLAB® on the Grid”, EGEE-TR-2008-001, 2008. *https://twiki.cern.ch/twiki/bin/view/EGEE/GLiteAndMATLABToolkitIntegration*

[1] Curtis F. Gerald, Patrich O. Wheatley, “Applied Numerical Analysis”,Addison Wesley, New York, 5nd edition, 1994.

[2] J. W. Thomas, “Numerical Partial Differential Equations”, Springer, New York, 1995.

[3] G. D. Smith, “Numerical Solution of Partial Differential Equations”, Clarendon Press, Oxford, 3nd edition, 1998.

HaDeS: a Scalable Service Oriented Deployment System for Large Scale Installations

Massimo Gaggero[1], Gianluigi Zanetti[2]
[1]CRS4, Edificio 1 - Pula (CA), Italy, {gianluigi.zanetti,massimo.gaggero}@crs4.it

***Abstract*—Building large computational facilities requires scalable and flexible deployment tools that can cope with massive loads. Classical installation methods are not very flexible, since they are usually limited in the number of OS supported, rely on transfer solutions that impose constraints on network topology, and do not scale very well. Here we describe HaDeS (Hardware Deployment System), a new deployment system for large scale installation designed to be agnostic with respect to the network topology and the OS deployed and to scale with the number of nodes being deployed.**

***Index Terms*—Deployment, scalability, operating system, hades, cybersar, metalink.**

I. Introduction

Cybersar [1] is a cyber–infrastructure that connects and supports all the research organizations in Sardinia that heavily employ computational facilities to support their research activities. Together with the "classical" use of HPC facilities, Cybersar supports research focused on the study and development of new computational paradigms, e.g., infrastructure as a service for scientific applications, that require continuous development of the entire software stack, from the OS, to libraries, middle-ware and applications. Cybersar infrastructure thus needs to support, in a scalable manner, heavy customization of Operating Systems and frequent cluster reconfiguration in a spirit similar to what is done in the project GRID5000 [2].

Building large computational facilities require scalable and flexible deployment tools that can cope with massive loads. Classical installation methods are usually limited in the number of OS supported and, typically, not very flexible. Moreover, they usually rely on broadcast transfers or remote OS mounting solutions that are not very robust, are network topology dependent, and do not scale very well with the number of nodes to be installed. On the other hand, to support cybersar research we needed a hardware deployment system that satisfies the following requirements: it should be

- independent from both the OS to be installed and the packaging system;
- fast and scalable with respect to the number of nodes;
- support a fully automatized and unattended deployment process;
- able to support incremental changes in large scale installation;
- integrable in complex system, e.g., so that it can be used as service component in SOA based work-flow applications,

Since the deploy system is supposed to be orchestrated by higher order entities [3], [4], it does not need to be a complete cluster management system, but just an intelligent and flexible deployment tool.

To satisfy these requirements we developed HaDeS. HaDeS image distribution mechanism is based on standard transfer protocols used in common p2p file-sharing applications. HaDeS can, therefore, work across LAN boundaries and it is compatible with multi-site geo-

graphical installations. HaDeS development has been kept strictly independent from the OS deployed and no assumptions were made about the images transferred. The system is built following SOA principles and the controlling interface is developed as Web Service. This approach allows for an easy integration of HaDeS as a module in complex management infrastructures while keeping it also usable as a standalone application. HaDeS is used within Cybersar as a component of the VIDA [4] infrastructure to manage fast whole cluster redeployment, e.g., to reconfigure clusters as hadoop (an open source implementation of map reduce [5]) machines, and to deploy test installations of cloud computing middleware such as OpenNebula [6].

A. Flexibility: OS Image Deployment Vs Package Installation

The automatic and unattended tools commonly used for cluster setup and configuration are packaging-system oriented and often strictly related to the OS installed [7] [8] and to the packaging system [9] used; albeit few, [10], are more flexible and can handle different OS and packaging system. While this approach is appropriate for the usual handling of computational facilities where the setup is done basically only once and the update is incremental, it is not particularly efficient in a quickly varying research environment focusing on the computational infrastructure itself.

Package installation is usually performed as a sequence of steps of widely varying duration: a target disk is selected, partitioned and formatted; software is transferred from the network to the target host and installed; the post-inst phase performs specific customizations and the installation of software that cannot be automatized. Since it is dependent on the amount of configuration and customization done, post-install can take longer than the installation phase and require manual intervention.

A more flexible approach is the deployment of a pre-built and pre-configured image of the operating system as a single file. As shown in fig 1 the mandatory steps for image deployment

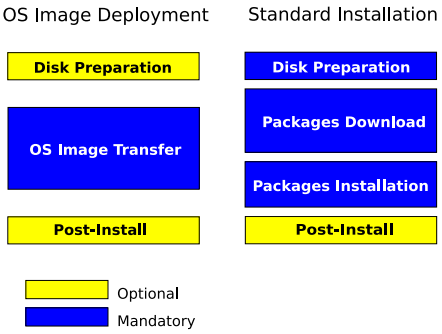


Fig. 1. Deployment and installation phases

are reduced to one. As the image comes with the whole file-system structure on it, partition formatting is no more required. Packages download and installation are replaced by the only operation of image file transfer from network to disk. A quick post-inst phase is possible, if required, but an accurate customization can avoid it. With current network and disk technology, the downloading of the disk images is bound by the transfer to disk bandwidth. Thus, with standard hardware, the time needed for a single image transfer is in the few minutes range.

B. Scalability: P2P and multi-protocol file transfer

Scalability is an issue related to the transfer protocol. Some tools deal with scalability using different file transfer protocols. SystemImager [11], for example, offers three different protocols with three different modules: rysnc transfer, multicast transfer and BitTorrent. Each one has advantages and disadvantages: Rsync is able to transfer only file deltas but it does not scale, multicast is more scalable but it requires specific network topology, BitTorrent can take advantage of multi-servers and downloaded chunk sharing but forces the images to be provided as torrent file. Moreover, since many ISPs and firewalls drop p2p protocols, geographical deployment cannot be done.

A flexible solution that overcomes these limitations is the Metalink File Format [12]. With Metalink it is possible to indicate different

sources and different transfer protocols pointing to the same file. Compatible clients can select between protocols and source to maximize the incoming bandwidth. Mixing together HTTP sources and P2P sharing for example, overcomes strict network policies while allowing for local chunks sharing.

C. Full Automation: controlling the hardware

Installation or deployment typically requires two reboots:

- machine power-on/reset;
- boot of the installation system;
- installation/deployment process;
- machine reboot.

In order to be really automatic and unattended, deployment has to manage also the hardware power switching. A standard embedded methods often available as standard is the IPMI protocol. As an alternative, many main-board manufacturer provide proprietary solution like HP Onboard Manager and iLO and Sun Ilom. One tool able to deal with hardware and issue power switching command is xCAT [13]: it uses IPMI standard to control power and boot device. However, it does not handle other Platform Control (PFC) methods and it does not provide full support for image deployment.

D. Integration

The usual tools for cluster management and provisioning tend to have an autocratic approach, i.e., the assume to be the only system controlling the computational infrastructure, do not usually adhere to interfacing standards and are very difficult to integrate in more complex systems. A more modern approach is to think to the deployment system as a component, that can work standalone, but it can also be readily integrated in a larger control-plane infrastructure, e.g., orchestrated as a SOA [3], [4].

II. Discussion

HaDeS is mainly a flexible tools for research: its development has been kept strictly independent from the OS deployed and no assumptions were made about the images transferred.

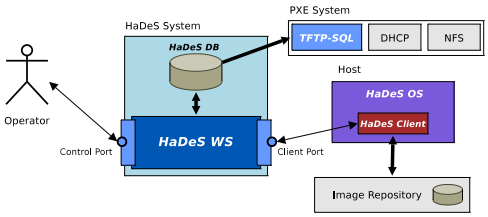


Fig. 2. HaDeS components

HaDeS system is composed by 5 different software components:

- the HaDeS Scout Image;
- the HaDeS Client;
- the HaDeS Web Service along with the HaDeS Database;
- the SQL-TFTP Server;
- the PFC Web Service.

A. HaDeS Scout Image and HaDeS Client

HaDeS works on the target host atop of a minimal disk-less operating system, the HaDeS Scout Image; as the target machine is booted, the OS is loaded via PXE. We are using for the Scout Image a Gentoo [14] LiveCD image, since it is small and contains a wide selecton of the disk and file system management tools needed by the HaDeS client. Moreover, the Gentoo LiveCD is able to detect a large quantity of different hardware devices and it can be very easily upgraded and customized using Gentoo tools such as Genkernel and Catalyst. The Scout Image is mounted from a NFS server by the target system and the root file-systems contained is loaded in memory, to reduce the network impact of NFS accesses. The NFS mounting of the Scout Image does not affect scalability, since only small and infrequent NFS data transfers are required during the boot and deployment processes. Moreover, the performance impact of NFS could be reduced, if needed, with server redundancy.

Once loaded, the Scout Image starts all the services of a common Linux system such as network configuration, SSH and Syslog daemons; the last service started is the HaDeS client. The latter collects information about the target host

and use them to query the HaDeS Web Service on the actions to be performed.

Then deployment starts following configuration information provided by the Web Service: the related schema is called *host setup* and it indicates the target partitions, the type of file-system to be created/deployed and the remote and local location of the image to be deployed. The host is then rebooted from a boot device at the end of deployment as per setup instructions.

During the process, HaDeS client talks to the HaDeS Web Service, reporting process status and errors occurred.

In order to adhere to well known standards, the setup is a XML file and communication between client and Web Service is implemented as SOAP messages. The client is written in Python 2.5 [15] and run atop of an instance of the Twisted networking engine [16].

B. HaDeS Web Service and Database

All the information about host managed by HaDeS are stored in the HaDeS Database. Hosts are described by their name, the setup to use for deployment, the next boot device, the collection of the network interfaces with MAC and IP address and all the Platform Control (PFC) methods provided to the host and which one is currently in use. The database is used by the HaDeS Web Service to provide clients with all the instructions needed to perform the deployment. Details about PFC methods are used, instead, by the PFC Web Service to control deployment targets hardware.

The HaDeS Web Service provides two ports:

- *Control Port:* it is intended to be used by the user, by means of the HaDeS tools, to communicate with the database requests such as list and details of hosts, setups, boot types, PFC details etc...;
- *Service Port:* it is the port used by the HaDeS clients to ask for instructions and to report process status.

When a client sends data gathered about a host that is not already in the database, and HaDeS is setup for automatic node discovery, the Service Port issues to the Control Port an add command for that host.

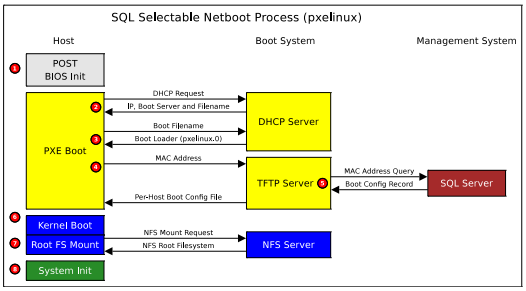


Fig. 3. SQL selectable PXE boot

HaDeS Web Service is developed in Python using the PyGridWare [17] library, a python implementation of the WS-Resource framework and part of the Globus [18] Toolkit. It is WSDL 1.1 [19] compliant and uses SOAP [20] messages. The database engine used is MySQL.

C. SQL-TFTP Server and PFC Web Service

Two important HaDeS software components not strictly related to deployment are the SQL-TFTP Server and the PFC Web Service. They control the network boot process when PXE is used by the nodes. They are crucial components of HaDeS since the deployment process requires hardware power control and automatic boot selection. The PXE boot system relies on a TFTP server to retrieve the boot loader and related configuration files, being able to programmatically select the boot file provided by the TFTP server makes it possible to control the nodes booting sequence.

We have developed a patch for the TFTP server that interface it to the SQL Database allowing boot selection just as a matter of changing the relevant boot record in the database when a deployment is issued to HaDeS, the Database makes the node point to the HaDeS scout image and, when finished, restores it back to the default device. Figure 3 illustrates the role of the SQL-TFTP server in selectable boot process.

The PFC Web Service instead is in charge of handling the way the hardware is turned on or

off or reset. The most reliable way is to issue power command using IMPI messaged to the node's Board Management Controller (BMC) or the equivalent for the board manufacturer (HP iLO for example or Sun iLOM). PFC Web Service acts as a proxy to these protocols: depending of the data provided by the HaDeS Database, it issues to the hardware the appropriate message when a reset is required to start the deployment. It is developed as a modular system allowing new protocols and new platform control methodologies to be added later.

D. Scalability

Scalability has been reached in HaDeS with the adoption of the *aria2* [21] utility developed by Tatsuhiro Tsujikawa. Aria2 is an application for the parallel download of files from multiple sources/protocols such as HTTP(S)/FTP and BitTorrent. While the data is downloaded using HTTP(S)/FTP, it is uploaded to the BitTorrent swarm. Thus, In this way just a small number of nodes directly downloads from the image repository while the others refer to them and one to each other using BitTorrent. This spreads the transfer to the internal LAN making it possible to use all the available aggregate bandwidth of the network infrastructures leaving a reasonable high and constant transfer speed per client instead of a $\frac{1}{N}$ drop, where N is the number of nodes.

III. Preliminary results

Several proofs of concept and functional tests have been carried out on HaDeS, while intensive efficiency and scalability tests are currently being performed. Here we describe some of the preliminary results thus far collected. Figure 5 illustrates results from a scalability test done deploying an image of about 3 GBytes over 16 nodes. The image is hosted in a single HTTP server with a 1Gbps link to the node's switch. Aria2 is used to download the image and share available chunks with BitTorrent. The "HTTP/BT" curve show the average transfer

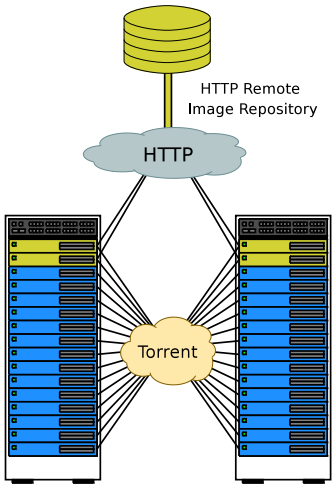


Fig. 4. Combinated use of HTTP and Bittorrent protocols

speed per client as the number of nodes increases, while the "HTTP" represents the estimated per-client speed using just HTTP. The use of the p2p sharing has led to an effective per node transfer rate of about 20 MBytes/sec, that is essentially constant with respect to the number of nodes deployed. As a reference, the effective disk bandwidth on the same node hardware is of about 40 Mbytes/sec.

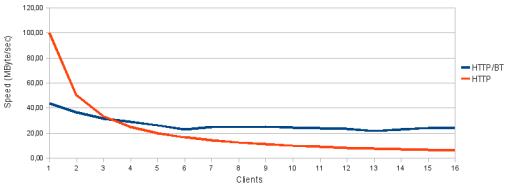


Fig. 5. Per-client transfer speed

The second test described is the deployment over 72 nodes of a 4,5 GBytes image during a Hadoop (an open source Map Reduce implementation [5]) evaluation test. As above, the image is downloaded by a single HTTP server and available chunks are shared using the BitTorrent protocol. Table I shows that the deployment time with hades is 323 sec, about 5 minutes, while the transfer only takes about 4 minutes; the difference of 75 seconds is the time

spent to create the accessories partition required by Hadoop to store data. The average transfer speed is about 19 MByte/sec, definitely better than the estimated 1,38 MByte/sec of a simple HTTP download from 72 clients.

TABLE I
DEPLOYMENT OF A 4,5 GBYTES IMAGE OVER 72 NODES

HaDeS Time	Aria2 Time	Transfer Speed
323,49 secs	248,47 secs	19,12 MByte/sec

IV. CONCLUSION

HaDeS has been developed initially with the aim to provide a flexible tool to allow the rapid deployment of heavily customized Operating System images and the fast switching of clusters between production and development environments. Scalability results from the use of p2p technologies such as BitTorrent protocol and Metalink file format. The Web Service interface allows HaDeS to be integrated as component in complex system SOA based. From this point of view, HaDeS has been successfully used in the VIDA [4] system for virtual cluster deployment. Other experiments and evaluations in the Cybersar project has been carried out using HaDeS as deployment tool and configuration switcher for the Cybersar cluster (72 nodes) hosted at CRS4. During Hadoop framework evaluations, HaDeS has switched the whole cluster from production to development and back to its initial state. The whole evaluation took less than two days, with just six minutes for the complete deployment of a pristine Hadoop environment and a minute to revert back the nodes into production. In a similar manner, HaDeS has been used to deploy a test installation of the OpenNebula cloud computing middle-ware [6]. Feedback from these tests has suggested a quantity of improvements and a new 2.0 version is under development. Changes include the development of new drivers for different OnBoard Control Systems, improvements in system robustness and resilience, deployment process monitoring and so on. At the same time, new test are planned to investigate scalability performance of HaDeS in different network topologies and its behavior at the geographical scale.

REFERENCES

[1] P. Anedda, C. Guidi, G. Zanetti, A. Bosin, A. Masoni, D. Mura, N. D'Amico, and I. Porceddu, "Cybersar: a new computational infrastructure for research in sardinia," in *3rd IEEE International Conference on e-Science and Grid Computing, Bangalore, India, 10-13 Dec. 2007*, December 2007.

[2] F. Cappello, E. Caron, M. Dayde, F. Desprez, E. Jeannot, Y. Jegou, S. Lanteri, J. Leduc, N. Melab, G. Mornet, R. Namyst, P. Primet, and O. Richard, "Grid'5000: a large scale, reconfigurable, controlable and monitorable Grid platform," in *Grid'2005 Workshop*. Seattle, USA: IEEE/ACM, November 13-14 2005.

[3] P. Anedda, S. Manca, M. Gaggero, and G. Zanetti, "Soa based control plane for virtual clusters," in *Euro-Par 2007 Workshops Parallel Processing - HPPC 2007, UNICORE Summit 2007, and VHPC 2007*, vol. 4854, 2007, pp. 154–163.

[4] P. Anedda, C. Guidi, and G. Zanetti., "Vida: A new virtual images deployment architecture," in *Workshop finale dei Progetti Grid del PON "Ricerca" 2000-2006 - Avviso 1575*, 2009.

[5] A. Bialecki, M. Cafarella, D. Cutting, and O. O'Malley, "Hadoop: a framework for running applications on large clusters built of commodity hardware, 2005," *Wiki at http://lucene.apache.org/hadoop*.

[6] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster, "Capacity leasing in cloud systems using the opennebula engine," in *CCA08: Cloud Computing and its Applications, Chicago Oct. 22/23 2008*, October 2008.

[7] C. Winter and D. Layfield, "Using solaris jumpstart with the solaris 10 os for x86/x64 platforms," *BigAdmin System Administration Portal*, March 2007. [Online]. Available: http://www.sun.com/bigadmin/features/articles/jumpstart_x86_x64.jsp

[8] A. Nashif and U. Gansert, *AutoYaST - Automatic Linux Installation and Configuration with YaST2*, SUSE Linux Products GmbH. [Online]. Available: http://www.suse.com/~ug/autoyast_doc/index.html

[9] "Anaconda/kickstart." [Online]. Available: http://fedoraproject.org/wiki/Anaconda/Kickstart

[10] T. Lange, "Fai - fully automatic installation." [Online]. Available: http://www.informatik.uni-koeln.de/fai/

[11] [Online]. Available: http://systemimager.org

[12] "Metalink - open standard/framework/file format." Wikipedia, the free encyclopedia. [Online]. Available: http://en.wikipedia.org/wiki/Metalink

[13] "xcat - extreme cluster administration toolkit." [Online]. Available: http://xcat.sourceforge.net/

[14] "Gentoo linux." [Online]. Available: http://www.gentoo.org/

[15] "Python programming language," Python Software Foundation. [Online]. Available: http://www.python.org/

[16] "Twisted matrix labs," WWW. [Online]. Available: http://twistedmatrix.com/trac
[17] "Python/pygridware," June 2007. [Online]. Available: http://dev.globus.org/wiki/PyGridWare
[18] "Globus toolkit." [Online]. Available: http://www.globus.org/toolkit/
[19] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, "Web services description language (wsdl) 1.1," W3C, March 2001. [Online]. Available: http://www.w3.org/TR/wsdl
[20] "Simple object access protocol (soap) 1.2," W3C. [Online]. Available: http://www.w3.org/TR/soap12-part1/
[21] T. Tsujikawa, *aria2c - The ultra fast download utility*, 1st ed., 07 2009. [Online]. Available: http://aria2.sourceforge.net/

EyeGrid, a webOS for Access to Distributed Resources

Gian Mario Mereu[1], Daniele Mura[1] and Daniele Siddi[1]
[1]Cosmolab Consortium, via Marengo 3, Cagliari, Italy
{*daniele.mura,gianmario.mereu,daniele.siddi*}*@ca.infn.it*

***Abstract*—`EyeGrid` is a web operating system which grants a user-friendly access to hardware and software resources, taken from grid or cloud computing. It allows a full grid use control, passing through a normal *user interface*.**
`EyeGrid` is based on EyeOS[1] and uses Web 2.0 technologies.
`EyeGrid` at the actual state supports the gLite[2] middleware. In the near future, we shall add interface with Globus[3], Storage Resource Broker[4] and the browsing to local file systems.

***Index Terms*—Cosmolab, Cybersar, eyeOS, grid computing, webOS.**

I. Introduction

THE goal of the present project is to acquire a modern *user interface*, which must be able to take care of the final user needings.
`EyeGrid` is a web operating system born to pursue that goal. It is based on the EyeOS operating system, an open source webOS written in PHP/Javascript, which uses the modern Ajax technology, fast and powerful and it has cross-browser capability. You don't need to install anything on your computer to use EyeOS.
`EyeGrid` environment has comprehension of most common utilities like text editors, some network clients, a file manager, and others.
`EyeGrid` can connect to an *user interface* and sends grid commands for processing. It can manage multiple jobs, generate jdl files, manage proxy certificates and realize input/output of jobs in a couple of clicks, freeing the user from having to learn and use annoying and complicated command-line instructions.
It can also give the user the possibility to launch interactive jobs.
An interactive job, typically window-based, runs on a *worker node* and requires the interaction of the user. Any program can be an interactive job, on condition that it's installed on the *worker node*. The execution of a job can be launched and controlled in different moments, leaving the user free to disconnect and return back later to retrieve his output.
Another important function is the integration between the gLite LFC/SE and the EyeOS file system.
This way you can copy and paste files or entire directories from/to the grid in a user friendly fashion.
The current release of the software is downloadable and usable from our web site, reachable at the address http://eyegrid.ca.infn.it

[1]http://www.eyeos.org
[2]http://www.glite.org
[3]http://www.globus.org
[4]http://www.sdsc.edu/srb/index.php

II. EyeGrid

`EyeGrid` runs on a web server and a database management system, and does not require other special software. Here follow a description of the software, analyzing the idea, the supported features and the implementing chooses.

A. Login and configuration

The access to the virtual operating system is managed by a typical login system. EyeOS itself has a complete user/group management system, and every user has a separate desktop with his

own files and folders. A number of concurrent applications can be used.
One of these is `EyeGrid`. Every user has his own entry in the support database and his own configuration.
To have grid access, the user must have an account on a gLite *user interface* and this has to be reachable from the web server with the SSH protocol, with SCP enabled.
The application has a configuration section which contains a list of accounts. These accounts are normal *user interface* accounts, which have to be pre-configured and completely working. It's also mandatory the presence of a valid SSL personal certificate.
When the user explicitly presses ***connect*** on an account row or does an action that requires the connection, he will be asked for a password, but just the first time he does it.
Further, the user will not be asked. The password is saved on a PHP session variable on the web server and used every time it is needed. At the end of the session or after disconnection, it will be destroyed.
Once connected on a *user interface*, a user can view, generate or destroy his proxy certificate. To build it, the personal certificate located on the *user interface* is taken. The user is asked for the the private key passphrase, if any, and for the role he wants to assume.
In the lower bar are shown the connection status and the proxy time still left.

B. Job management

The job management interface is composed of a job list in a table form, where each row corresponds to a job, and a message box where every output message appears.
The user has a simple job list that can be used to manipulate every single job. The informations about the job are sent to a DBMS[5]
. There is no limit on the number of jobs that can be looked up, and they can be sorted within a couple of keystrokes.

[5]currently, MySQL 5 are used

The commands can be given using the top toolbar, which has some buttons linked to particular actions.

1) The job resume panel: this panel contains all the informations about jobs. For each job, the jdl file name, its status, submission time and refresh time are indicated. For interactive jobs there is a specific field usable to open the interactive job window, which is described later.

2) The job detail view: opened by double-clicking on a row. It shows the gLite ID, the status with a reason string, submission/refresh time and the entire submitted jdl, if found on the eyeOS file system.
In the bottom-left corner, the user can read and write specific job notes.

3) The jdl generation form: it is one of the most important features of the program.
The panel is divided in two frames. The left frame contains some text fields and other input objects. The right one contains an editable text area. The jdl text will be generated inside it.
A skilled user can write by hand or paste his jdl in this area, without using the generation utility. He can also upload a file and open it with the proper button.
If he chooses to use the objects in the left side, he can fill up some information about the jdl: what command he wants to execute, its parameters, input and output files and any additional requirements needed etc.
By clicking the ***Generation*** button the user gets his jdl. This simple procedure frees the user from learning the jdl language.
All the paths in the *input sandbox* are relatives to the local eyeOS file system. This means that the user has to upload files in the eyeOS or in the LFC before submitting the job.
By clicking the ***Submit*** button, the jdl file is parsed, all files named in the input sandbox are included in a folder and then compressed in an archive. The jdl then is re-written with the files paths relatives to the new structure, and included in the compressed archive. Now, the archive is ready to be transferred to the UI and submitted.
This is done, by using the SCP protocol, to

the home directory of the user, and a proper command is launched. This command checks whether a *computing element* that can accomplish the request is present or not. If the check succeeds, the submit command is launched.
If the submission is successful, a new row will appear in the list.

C. Interactive jobs

The interactive jobs feature gives the possibility to launch graphic applications on the grid, using grid computing power and grid storage space without the restriction to use them just with command-line programs.
Once an application set is chosen and included in the *interactive job* list, the grid admin has to install it on every involved *worker node*, with a bit of additional software (an X server, a window manager and a VNC server).
Then the user has the possibility to click on the name of the software, optionally to pass an input file to it, and a special jdl file will be submitted. This job executes a custom pre-compiled script wrapper. When the job will go in *Running* status, the job row will be refreshed with the adding of a clickable special field, which causes the opening of a window, when clicked. The window contains a Java VNC applet which connects to the *worker node* and shows the requested software.
In this situation, a possible problem is to find a free *worker node* when we want to use interactive applications. The grid can be busy with batch jobs, for example.
A solution to this problem is to preserve an *ad-hoc* queue, and some *worker nodes* to manage exclusively these kind of jobs; another possible way is to implement priority policies which grant full access to resources to this class of jobs.
The important innovation is the possibility for the user to graphically launch a computing process on the grid, and then close the `EyeGrid` session without stopping it. This procedure can be repeated until the interactive application is running. When the application is closed, the job goes in *Done* status and the resources are released.

D. File system browsing

EyeOS has a file system navigation utility.
It has an infrastructure that supports different home directories per user, and a permission system.
If usable through normal command-line commands, any storage system or extern file system can be integrated in `EyeGrid`. For simplicity, we will refer to them like "*file systems*" later in this paper.
Actually, the LFC catalog, normally accessible only through a gLite *user interface*, is fully working.
This feature allows the user to copy and paste files and directories from and to the catalog with extreme ease, in a visual environment.
To achieve this goal, the technique used was the same as above: the actions executed by the user are translated to corresponding command-line strings and sent to the *user interface*.
Cfr. the technical section for implementation details.

III. Technical information

The entire architecture of the software is built around a paradigm Ajax-oriented: the server side uses PHP 5 and the client side normal Javascript. The two sides intercommunicate using XML formatted data.
EyeOS is built using the event-driven paradigm, and it is possible to execute code from one side to the other with relative ease, triggered by any event.
The EyeOS code was deeply modified to permit the use for `EyeGrid`. A wrapper service was added to perform the LFC catalog browsing.
In detail, we have built a script capable to mirror the LFC structure in a known directory on the web server. The script has the possibility to mirror an entire catalog or to synchronize just local updates.
The browsing, for speed reasons, is redirected on the mirror. When an action (copy, delete,

directory creation etc.) is performed, a shell command is generated and submitted to the UI through the SSH connection. Therefore, the output is captured and the proper action is executed.
The synchronization from the catalog to the mirror is realized by doing a little update on the LFC server. This has a DBMS that catalog the files presents on the distributed storage elements[6]. When a user does any action (by `EyeGrid` or with any UI) that influences the catalog, this event causes an update on a specific database table.
In that table, we added a trigger that reconstructs the differences and updates the mirror, by calling the script updater named above.
A complicated SQL procedure, and a third-party library allows one to send shell commands from the DBMS.
That implies a strong connection between LFC server and the web server: the *mysql user* from the LFC server has to send a command to the *apache user* on the webserver, because `EyeGrid` runs with that permissions on the web server. This solution is applicable when web and LFC server runs on the same LAN. In other cases, this can be a security risk.
However, you can disable the mirroring anytime and directly browse the catalog, without the need to update anything, but the price is slower access to the catalog.

IV. Conclusions

The goal of this paper is to make the Grid access simpler, and to free the users from learning the powerful but complicated Grid language.
It can be useful for students, researchers and enterprises which need to have an immediate, distributed, ready to use Grid front-end.
Future development can integrate remote storage systems different from LFC/SE and manage also parametric jobs, which will group similar jobs and treat them with some special functions. Another important level of generalization will be reached when integration with Globus will be completed.

[6]this step requires the upgrade to MySql version 5.x, which supports the triggering

Acknowledgment

The authors would like to thank the **Cosmolab Consortium**.
`EyeGrid` is realized for **Cybersar**, a project co-funded by the Italian Ministry of University and Research within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione".

Daniele Mura Daniele Mura has a PhD in physics, works in the Cosmolab organization and is in charge for a specific objective on the Cybersar project

Gian Mario Mereu Gian Mario Mereu has a bachelor in Computer Science and works primarily as a system administrator for the Cosmolab organization

Daniele Siddi Daniele Siddi has a bachelor in Computer Science and works primarily as a software developer for the Cosmolab organization

Improving Node Inter-cluster Point-to-point Communication by using the UDP Protocol and NAT Service

Francesco Gregoretti[1],Gennaro Oliva[1] and Carlo Maurizio Santoro[2]
[1]*Institute of High Performance Computing and Networking ICAR-CNR, Naples, Italy, francesco.gregoretti@na.icar.cnr.it, oliva.g@na.icar.cnr.it*
[2]*University Parthenope, Naples, Italy, carlomauriziosantoro@gmail.com*

***Abstract*—In the context of multi-site parallel applications where a single MPI program is executed on several clusters linked with grid technologies (i.e. the Globus toolkit and MPICH-G2) a problem can arise in the presence of clusters with private networks. In such an environment two nodes belonging to different clusters with no public IP address cannot communicate directly. We have seen how this problem can be solved in a portable manner by introducing user MPI processes, running on the front-end, whose only purpose is to route messages among clusters. In this work we have improved private node inter-cluster point-to-point communication by using the UDP protocol and a NAT service on the front-end. The first enhancement improves network performance by reducing the transport protocol overheads while the second enhancement avoids useless memory copies for the message on the forwarding host. Performance comparison is presented with the user MPI process solution and the impact of the two enhancements is separately evaluated.**

***Index Terms*—MPI, Grid.**

I. Introduction

A Multi-site parallel application [2] is composed of multiple processes running on different clusters that may need to exchange data during execution. A robust and productive development environment for this class of applications is the MPICH-G2 [6] library based on the Globus Toolkit Grid Technology.

Using this library in the presence of clusters with private networks can be problematic because in such an environment two processes, running on two nodes belonging to different clusters with no public IP address, cannot communicate directly.

We showed [3] how this problem can be solved in a portable manner by introducing user MPI processes (termed forwarders), running on the front-end, whose only purpose is to route messages amongst clusters. To enable transparent use of forwarders we have developed an MPI library whose communication primitives uses forwarders when needed.

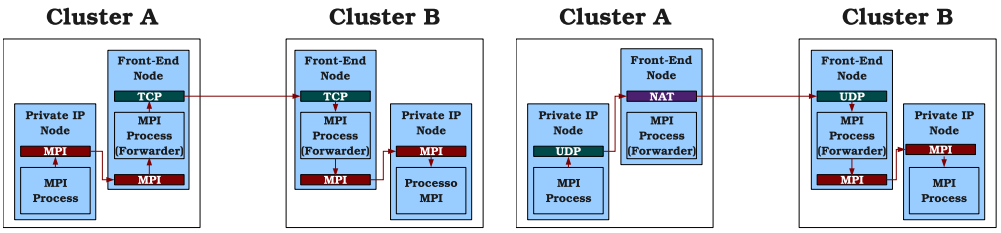
The introduction of forwarders has also improved performance of collective operations by using a delegating mechanism [4].

In this paper we describe how we designed, implemented and tested a new private node inter-cluster communication strategy to improve point-to-point performance. This strategy has two major improvements with the respect to the one depicted in figure 1a:

- it uses the UDP in order to minimize transport protocol overhead and
- uses the NAT service in order to avoid useless memory copies of the message on the forwarding host.

This solution can be used in conjunction with the old one as it inherits all its portability and can exploit the presence of a NAT service on the front-end when available.

We developed a new implementation of the MGF point-to-point communication primitives



(a) Old design (b) New design

Fig. 1: Implementation design comparison

that uses NAT and UDP and executed some tests to evaluate performance improvements by comparing the new implementation to the old one to understand how much advantage comes from the usage of NAT and how much from the usage of UDP.

II. New point-to-point communication implementation

The new point-to-point communication has been implemented by replacing one forwarder with a better low-level forwarding service provided by the operating system kernel and by reducing the transport protocol overhead using UDP instead of TCP.

A. Cluster networks

Generally there are two options for the network connection of a cluster:

- to provide all the cluster nodes with a public IP address and connect them to the WAN, that is normally done only for small size cluster;
- to connect a single node (called front-end) to the WAN with a single public IP address and to connect all nodes, including the front-end, with a private TCP/IP internal network that is more commonly adopted.

Forwarders are user processes executed on the front-end nodes of clusters with private networks. When two MPI processes, executing on different clusters with private networks, namely A and B, need to communicate, the sender sends its message to the forwarder A using the native MPI library; forwarder A sends the message via TCP to forwarder B that finally takes care of message delivery to the receiver (figure 1a) using the local MPI library.

B. Replacing forwarders with NAT

The front-end of a cluster with private network usually provides access to WAN for the other nodes, acting as a gateway, using a Network Address Translation (NAT) service. NAT is a service used to hide private networks behind a single public IP address. It needs one host interconnected to both the private and the public network, that forwards outgoing packets from the hosts of the private network to the hosts of the public network by replacing the address in each IP packet with its own public IP and by keeping track of all connections to be able to forward replies to the right internal host. NAT can also be used for incoming communications but the receiving nodes and ports must be known in advance. Linux like many other modern operating system provides a NAT service at the kernel-level.

In our new implementation (figure 1b) the NAT service running on the front-end of cluster A replaces the role the forwarder A, while forwarder B is still needed for message delivery. Using a kernel level NAT service improves performance in forwarding since, by acting on the IP level in kernel-space, it avoids the TCP/IP

stack traversals and the context switching from kernel to user mode and vice versa.

C. The usage of UDP

The second improvement introduced in the new implementation is the usage of the UDP protocol. We decided to use UDP because of its lower overhead and higher transmission speed with the respect to the TCP protocol.

Unfortunately UDP is unreliable: it provides no guarantees for datagram delivery, no protection from duplication and have a limit for datagram size. Since MPI communications need to be reliable and don't have message size limits, we needed to implement a robust and reliable point-to-point communication using the UDP protocol.

To solve datagram size limitation, large messages are split in multiple numbered datagram and sent to the forwarder using UDP. For each message the forwarder receives and stores all its datagrams, reassemble the message and sends it to destination.

To detect and correct transmission errors we used a simple stop-and-wait method where the sender waits an ACK (acknowledgement) by the receiver before sending the next datagram. If the ACK does not reach the sender before a certain time, known as the timeout, the sender sends the same datagram again. The duration of the timeout increases with simple exponential back-off algorithm.

III. Experimental Results

We did some execution tests to evaluate the new point-to-point primitives implementation with the respect to the old implementation. To evaluate how much improvements derived from NAT and how much from the UDP we implemented a TCP version of the communication routines that uses NAT and compared its performances, with the UDP version.

As a testbed we used two Beowulf class clusters: Vega (20 nodes Pentium 4 with Fast Ethernet), Beocomp (16 nodes Pentium II with Fast Ethernet), whose front-end nodes are provided with a public IP address.

Both clusters run Red Hat Linux 7.3, with kernel 2.4.20 and GNU libc 2.2.4. The local version of MPI is MPICH 1.2.5.2 with the ch_p4 device. The Globus Toolkit 2.4 is used as Grid middleware and the version of MPICH-G2 is 1.2.5.2. Vega is located at the Institute of High Performance Computing and Networking (ICAR-CNR), while Beocomp is located on the same LAN at the Department of Mathematics and Applications "R. Caccioppoli" of the University of Naples Federico II.

The test used is a simple ping-pong program, available in the MPICH test suite, where messages of different dimensions bounce back an forth between two given processes. A cycle of 100 bounces was executed for each message size and the total amount of time was divided by 100. Results obtained are represented in figure 2 where is evident that the TCP implementation that uses NAT performs better than the original implementation and that the UDP version is better that the TCP.

IV. Conclusions and Future Work

We have designed, implemented and tested new private node inter-cluster point-to-point communication primitives for the MGF library. We used the UDP protocol that improves network performance by reducing the transport protocol overhead and the NAT service to avoids useless memory copies for the message on the forwarding host. Tests executed have shown a good performance improvement and that both innovations contributed to the enhancement. The use of a listening UDP socket on the forwarder will be used to develop a multicast implementation of the collective communications to improve performances.

Acknowledgment

The authors would like to thank Sara Geneletti of Imperial College London for valuable comments on the manuscript

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

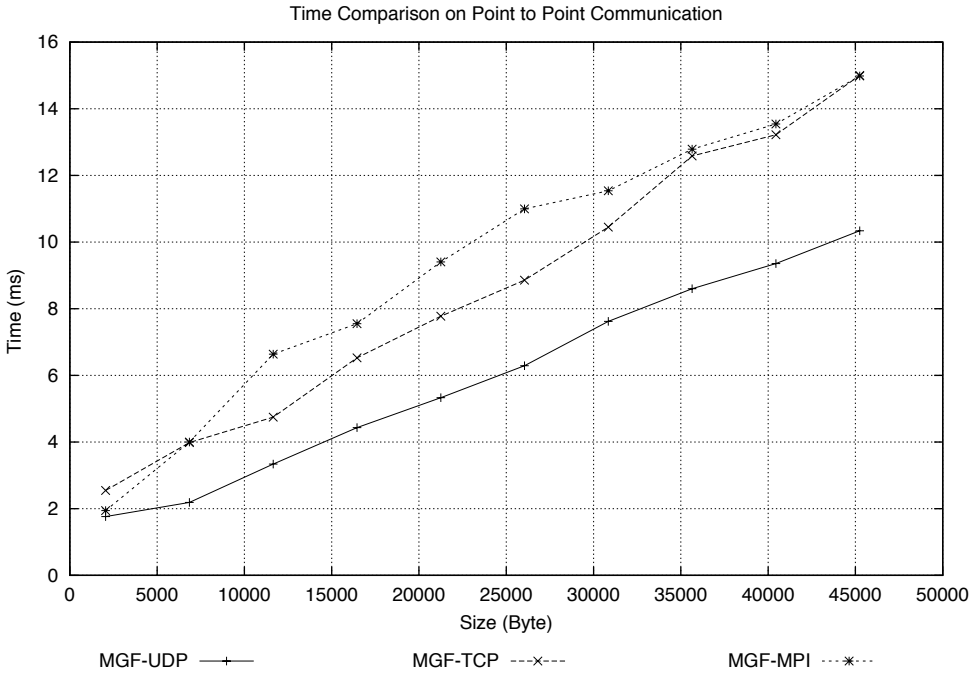


Fig. 2: Point-to-point performance comparison

[2] J. Frey, F. Gregoretti, G. Laccetti, A. Murli and G. Oliva, *"Multi-site Jobs Management System (MJMS): A tool to manage multi-site MPI Applications Execution in Grid Environment"*, Proceedings of the HPDC'15 Workshop on HPC Grid programming Environments and Components (HPC-GECO/CompFrame), IEEE Computer Society, 2006.

[3] F. Gregoretti, G. Laccetti, A. Murli, G. Oliva, U. Scafuri, *"MGF: a grid-enabled MPI library"*, Journal of Future Generation Computer Systems (FGCS), special issue on Grid and MPI, Elsevier Volume 24, Issue 2, February 2008, Pages 158-165.

[4] F. Gregoretti, G. Laccetti, A. Murli, G. Oliva, U. Scafuri, *"MGF: a grid-enabled MPI library with a delegating mechanism to improve collective operations"*, Proceedings of the 12th European PVM/MPI Users' Group Meeting (Euro PVM/MPI 2005), Lecture Notes in Computer Science, Springer, 2005.

[5] I. Foster, C. Kesselman, Computational grids, in: I. Foster, C. Kesselman (Eds.), *"The Grid: Blueprint for a New Computing Infrastructure"*, Morgan Kaufmann, 1998, pp. 15–51.

[6] N. T. Karonis, B. Toonen, I. Foster, *"Mpich-g2: a grid-enabled implementation of the message passing interface"*, J. Parallel Distrib. Comput. 63 (5) (2003) 551–563.

[7] M. Mller and M. Hess and E. Gabriel, *"Grid enabled MPI solutions for clusters"*, In Proc. 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2003), Los Alamitos, 2003. IEEE Computer Society, 2003, 18–25.

[8] S. Choi1, K. Park, S. Han, S. Park, O. Kwon, Y. Kim and H. Park, *"An NAT-Based Communication Relay Scheme for Private-IP-Enabled MPI over Grid Environments"*, in: Proc. of International Conference on Computational Science, ICCS04, Krakv, Polland, 2004, 499–502.

Francesco Gregoretti received his Ph.D. in applied mathematics and computer science from the University of Naples Federico II in 2006, his doctoral research being about multi-site parallel applications on the grid. Since 2001 he has had a permanent position as technician at the Italian National Research Council (ICAR-CNR). Dr Gregoretti's research focuses are in areas of parallel, distributed and grid computing.

Gennaro Oliva received a degree in mathematics in 1999 and Ph.D. degrees in applied mathematics and computer science in 2004 from the University of Naples Federico II. Since 2001 he has a permanent position as technologist at the Italian National Research Council (ICAR-CNR). Dr Oliva's research focuses on parallel and distributed computing; his recent work concerns multi-site application management on the grid.

Carlo Maurizio Santoro received a degree in computer science in 2007 from the University of Naples Parthenope. Since 2008 he works in Accenture Technology Solutions - Naples Delivery Center as a technical leader responsible for designing and maintaining the Business Intelligence infrastructure for the enterprise, including the data warehouse, reporting tools and data integration tools.

An Infrastructure supporting QoS Parameters for Composite Services Running on a Grid Environment

A. Di Stefano[1,3], G. Morana[1], G. Pappalardo[2,3], E. Tramontana[2,3], D. Zito[3]
[1]*Dipartimento di Ingegneria Informatica e Telecomunicazioni, University of Catania, Italy*
[2]*Dipartimento di Matematica e Informatica, University of Catania, Italy*
[3]*Consorzio COMETA, Via S.Sofia 64, I-95123 Catania, Italy*
{adistefano, gmorana, dzito}@diit.unict.it,{pappalardo, tramontana}@dmi.unict.it

***Abstract*—Services available over a Grid environment are characterised by a set of QoS parameters, e.g. desired response time, cost, etc., that are then employed as part of a service level agreement (SLA) between service providers and users.**

In the development of distributed applications, so-called "mashups" of services are continually gaining relevance, for they allow users to easily connect simple services in customised ways, thus giving rise to high level, hence more useful, composite services.

Currently, Grid users are endowed with facilities that allow SLA agreements to be negotiated for simple services, however no equivalent support exists for a composite service. In such a context, users should cope with such difficulties as: determining the overall QoS value of a parameter for an arbitrary combination of services; and smoothly defining, for component services, SLA agreements incorporating suitable provisions for specifying service recovery measures. These would allow failures of a component service to be handled gracefully, in such a way that the overall execution of the mashup would remain successful notewithstanding the failure.

In order to solve the said difficulties, we propose an infrastructure for building service mashups that provides users with automated support for: (i) calculating resulting QoS parameters, (ii) building SLAs for each component service and for the overall mashup, and (iii) providing the necessary runtime coordination for executing the component services in a suitably orchestrated way. It is also worth noting that the proposed infrastructure feature the ability to deal with component services that are available as Web services.

***Index Terms*—QoS, SLA, service composition, gLite.**

I. INTRODUCTION

NOWADAYS Grid middlewares are evolving toward the ability to support provision of services characterized by an high degree of complexity, in order to better satisfy the requirements of an increasing number of heterogeneous users. However, the current management of a single service only is becoming too much a restriction for users, because, during their work with Grid middlewares, users generally require the execution of *composite services* (or *"mashups"* of services), i.e. a set of different services, each one with a specific QoS level, whose execution is mixed together, according to some relationship of mutual interdependence.

Consequently, for the development of distributed applications, mashups are continually gaining relevance in that these allow users to easily connect simple services in customized ways, thus making an high-level, and so more useful, composite service to emerge. The existent Grid middlewares provide support for the

issues related to the negotiation of QoS for a single service, even when the service belongs to an administrative domain that is different from that of the user, however currently Grid middlewares lack of solutions to manage QoS provision and negotiation for composite services.

An infrastructure for managing the QoS for a composite service should provide the following support. Firstly, support to define a set of roles through which component services are to be connected has to be provided. Secondly, issues related with the negotiation of the QoS parameters between the final user and every component service have to be handled. Thirdly, the QoS pattern for an arbitrary configuration of services should be appropriately expressed. Fourthly, a supervision has to be provided for the SLAs to be drawn between the final user and the providers of the component services. Finally, monitoring and periodic checking of the abidance of the QoS bonds agreed within the SLAs should be available.

In order to provide the said support, our proposed infrastructure for building mashups of services supply users with automated means for: calculating overall QoS parameters, building SLAs for each simple service and for the overall mashup, and coordinating at runtime the execution of component services. The proposed approach realizes service composition in two phases: 1) a discovery phase that the consumer engages for each required service with specific QoS parameters and 2) a composition phase in which the consumer sets his/her preferences for the execution of the chosen services.

The services composition is managed in a coordinated way by the infrastructure described here. The infrastructure orderly negotiates the SLAs with each selected service provider, produces a global SLA taking into account the overall QoS and proposes it to the parties involved (i.e. the final user and the service's providers).

The paper is organized as follows. Section II focuses on the relationship between the SLA and the QoS. Section III analyzes the issues related with the composition of services in a Grid environment. The following Section IV describes the infrastructure by defining the components, outlining their activities and taking care of the issue of SLA management. Finally, Section V draws some conclusions.

II. QoS AND SLA

Quality of Service (QoS) is the ability to provide different priorities to different applications, services, users, or data flows, so as to guarantee a certain level of performance. QoS is related to the values of some non-functional parameters, such as e.g. response time, reliability, availability and cost [3], [4], [5]. Since QoS gives users the support for choosing the best one among different versions of a service on the basis of their expectations, it represents a crucial aspect in the development of actual Grid middlewares that can currently provide services only in a *best effort* fashion. The opportunity to supply services with several degrees of guarantee allows the Grid middleware to evolve towards the concept of *services' marketplace*.

In order to support QoS, the necessary coordination between service providers and consumers is formalized through a Service Level Agreement (SLA), i.e. a negotiated agreement between the two parties that records a common understanding about services, priorities, responsibilities, guarantees and warranties. For each service agreed, the SLA specifies the levels of availability, serviceability, performance, operation, or other attributes of the service such as e.g. billing. There are many specification of SLA (i.e. WSLA [12], WS-agreement [14], SLAng [13]), each one taking care of some characteristics of the context where SLA is used.

Generally, an SLA implementation should contain two main sections. The first section is related with the technical specifications of the target service and includes:

- the service name
- a service description, in terms of input/output parameters and availability

- the involved participants: provider, consumer and any *third party entities* to ensure a "trusted" service
- the service access mode, i.e. involved protocols and exchanged messages.

The second section takes care of all the non-functional parameters related to the service and includes:

- the service cost, according with a pricing policy
- QoS section, i.e. QoS levels specifically related to the provided service, number of involved parameters and the values/range allowed for each parameter
- exceptions/benefits, i.e. corrective actions to be undertaken when the agreements are violated, penalties related to a fault or potential price cost rise in case of advance target reaching. It is important to note that the 'agreement' relates to the services the customer receives, and not how the service provider delivers that service.

The process of creation of an SLA is usually anticipated by the exchange of a predefined template between the two parties. The template contains the customable aspects of the agreement and the roles by means of which it is possible to create the agreement itself. The creation of this template could be started either by service consumer or service provider.

III. Infrastructure for supporting QoS of Composite Services

Considering only the SLA management of a single service is often too much restrictive, because, during the normal use of the Grid resources, users refer to a set of services with some relationship of mutual interdependence, where each service has a specific QoS level. Consequently, the user has to consider the QoS of a whole *composite service* rather than the one related to a single service. Such *composite services* are closer to what the user wants to perform.

A composite service consists of several services connected with each other according to a direct graph, where the nodes represent elementary services and the edges represent the dependency links between two connected functions [11]. The dependency link indicates that the output of one or more services is used as the input by its (or their) successor, so as one of the services is a data source and the other a data sink. In such a graph, services are unaware of each other, i.e. services need not know predecessor or successor services.

Each component service declares a set of QoS parameters that the service provider has to guarantee, otherwise such a provider incurs in some penalties. The overall QoS of the composite service can be considered as a combination of the corresponding QoS of the constituting services, i.e.:

$$QoS(comp) = f(QoS(s_1), ..., QoS(s_n))$$

where: $QoS(s)$ is the set of QoS parameters for a service s, $comp$ is the composite service and s_1, s_n are the component services constituting $comp$. The main issue to face in the measure of the $QoS(comp)$ is the determination of composition functions, since it varies according to the QoS parameter considered and the rules of the composition. The same parameter can be expressed in several ways, on the basis of the composition network: e.g. in case of component services that run sequentially, the response time of the composite service depends on the sum of the response times of each component service. The overal response time can be expressed as:

$$QoS|_{rtime}(comp) = \sum_i QoS|_{rtime}(s_i)$$

while, in case of component services running concurrently, the overall response time becomes:

$$QoS|_{rtime}(composite) = \max_i QoS|_{rtime}(s_i)$$

A composite service, therefore, is characterised by: (i) a list of component services, (ii) the edges that connect such services, and (iii) a set of values for QoS parameters, that represent the QoS of the composite service.

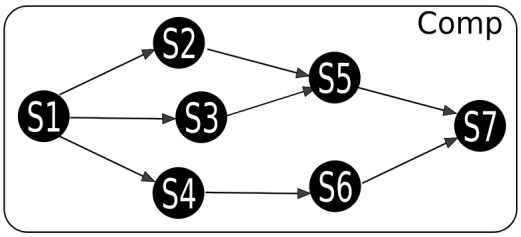


Fig. 1. Example of a composite service

IV. Infrastructure for managing Composite Services in a Grid Environment

A. Components at a glance

In order to support SLA and runtime coordination for executing composite services, the infrastructure presented here focuses on: (i) service discovery and service selection, (ii) coordination of the execution of a composite service and (iii) monitoring of the execution of every component service.

The proposed infrastructure consists of three main components: Controller, Receiver and Remon. Controller is the core of the infrastucture. It performs queries to the Resource Broker for discovering services and selects, among those found, the services that will constitute the composite service. During selection, Controller takes care of calculating the overall QoS (see Section III) and that QoS parameters satisfy the values required by the user and documented as an SLA. Once component services are selected, Controller sends, at the appropriate times, data and commands to these services to control their execution. Finally, Controller monitors and checks component services to ensure that their execution is as expected, according to the SLA.

Receiver plays the role of *executor* for the choices taken by Controller. Receiver is able to perform all the actions related to service execution: start, suspension, termination, checkpointing, abort, etc. For this, Receiver is connected with the Local Resource Manager (LRM) hosted on the Computing Element (CE), i.e. a set of computing resources available for executing services.

Remon has a twofold role: on one hand it monitors the runtime execution of the services using an appropriate reflective software architecture [9], [10]. Remon periodically checks whether intermediate deadlines, which have been estimated on the basis of the final deadline, are being missed. If these deadlines could not be reached, it starts some recovery activities to adjust the parameters of the underlying platform, so as to meet the following deadlines. On the other hand, Remon calculates and checks the service level parameters for the whole composite service.

At the CE tier, the infrastructure uses the Advance Reservation module (described in [8]) to perform the advance reservation and allocation of the resources involved in the composition.

B. Activities

The management of the QoS for a composite service consists of four phases:

- discover and select the services that match the users request (in term of QoS)
- combine QoS parameters that each service exposes, so as to satisfy the overall QoS requested by the final user
- coordinate SLAs of component services, so as to reach a global SLA that takes care of all the issues related with the user request
- launch and monitor the execution of the component services on each relevant resource.

In the proposed infrastructure, the initiator of the composition is Controller that starts the discovery phase by querying the Resource Broker for the component services that potentially contribute for the composite service. Generally, Controller retrieves multiple versions of each component service and selects among the potential services the one that ranks as the best, according to a function based on a weighted product model [15]. Once every service has been selected, Controller calculates the QoS values for the composite service, according to

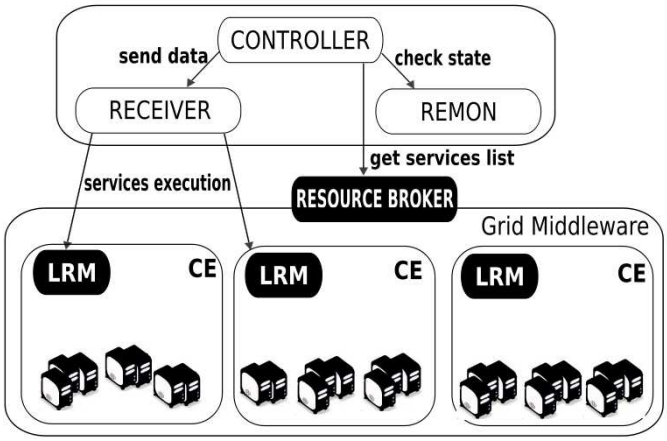


Fig. 2. Components of the infrastructure

the formulae discussed in III, and, on that basis, sets the price for the composite service. When the resulting price exceeds the price that user is willing to pay, Controller considers the second best version of a component service, so as to reduce the total cost. Afterwards, Controller checks whether the current conditions of the resources allow the execution of the composite service. When execution is possible, Controller starts the stipulation of the SLA among the service consumer and the service providers, otherwise it starts a phase of recalculation of the QoS parameters, in which Controller looks for an alternative set of values for the QoS parameters and proposes it to the user. In this last case, the user is asked whether the counter-proposed QoS levels are satisfying before execution can go on.

Once the overall QoS level for the composite service has been agreed by providers and consumer, Controller sends commands to Receiver for starting the execution of each services on the basis of the agreed behaviour. Receiver in due time will provide the appropriate input data for any service (stage in). Furthermore, Controller takes care of the data flow within the composite service, i.e. moving data from one service of the chain to the others, according to the rules of composition. During the execution of the service Remon acts as a local monitor and alerts Controller at the termination of the service, so as to allow Controller to move output data in a timely way (stage out).

C. SLA management

The execution of a composite service according to a set of QoS parameters implies the coordination among the service providers that have to respect the agreement drawn with the final user. All the service related data (in terms of functional and QoS parameters) are collected in a SLA that Controller sends to each provider involved in the negotiation process. Controller acts as trusted mediator between the involved parties. First, it discovers the provider for each service that will constitute the composite service, then it negotiates with each of this provider all the QoS parameters. When all negotiations are finished, it sends to each provider an SLA, where the results obtained from the previous negotiations have been recorded.

According to the schema described above, each SLA holds only the data related to the specific service provider. Each provider accepting the SLA signs it and sends it back to Controller. Once all SLAs have been received,

Controller builds a global SLA, sends it to the user, and waits the response of user. The execution of the composite service will start only at the reception of the signed SLA from the user.

The scope of these two categories of SLA are different: while single SLAs are local because each provider has only the Controller as counter-party, the global SLA aggregates data for the composite service and focuses on the final user. The global SLA in fact contains the whole agreement and lists all the hosts involved in the composition, together with their roles in the composition. All the technical data of the composite service are contained in the global SLA, whereas the detail of each single service execution are hidden to client.

V. Conclusions

The increasing focus on the issues related with the provision of complex services within the existent Grid middlewares, so as satisfy the requirement of an audience more and more heterogeneus, creates the need to handle the QoS assurances for composite services, rather than simple ones.

In this work, we have proposed an approach for automatically calculating the value of each QoS parameter for a composite service, consisting of lower level services connected among each other. Moreover, we have proposed a software infrastructure supplying means so that at runtime the needed provision of resources within a Grid environment allow QoS parameters to be satisfied.

For this, the proposed infrastructure performs several activities, i.e. discovers the resources needed for the composite service, selects services whose characteristics match user requirements, calculates the QoS for the composite service, coordinates the SLAs between parties involved in the negotiation and, finally, starts and monitors the execution of each component service.

References

[1] J. Hathaway, *Service level agreements: keeping a rein expectations*, in SIGUCCS, USA:ACM, 1995.

[2] L. Jin and V. Machiraju and A. Sahai, *Analysis on Service Level Agreement of Web Services*, In Hewlett Packard Laboratories (HPL-2002-180), July 2002.

[3] ITU-T (CCITT), Recommendation, *Terms and Definitions Related to The Quality of Service and Network Performance including Dependability*, August 1994.

[4] A. Bhargava and B. Bhargava, *Measurements and Quality of Service Issues in Electronic Commerce Software*, In Proceedings of International Conference on Application-Specific Software Engineering and Technology (ASSET), Texas, March 1999, IEEE.

[5] J. A. Zinky, D. E. Bakken, and R. D. Schantz. *Architectural Support for Quality of Service for CORBA Objects. Theory and Practice of Object Systems*, January 1997.

[6] G. von Bochmann and A. Hafid, *Some principles for quality of service management*, In Distributed Systems Engineering, vol 4., numb. 1, 1997.

[7] EGEE, *gLite HomePage*, `http://glite.web.cern.ch/glite/`

[8] A. Di Stefano and G. Morana and G. Scuderi and D. Zito, *Negotation-based Advance Reservation to manage SLA in gLite*, Final Workshop of Grid Projects, "PON Ricerca 2000-2006, Avviso 1575", March 2009.

[9] A. Di Stefano and M. Fargetta and G. Pappalardo and E. Tramontana, *An Infrastructure for Runtime Evolution of Software Systems*, In Proceedings of the Symposyum on Computers and Communications (ISCC), volume 2, pages 1129-1135. Alexandria, Egypt, June 2004, IEEE.

[10] M. Milazzo and G. Pappalardo and E. Tramontana and G. Ursino, *Handling Run-time Updates in Distributed Applications*. In Proceedings of Symposyum on Applied Computing (SAC), pp 1375-1380, Santa Fe, New Mexico, USA, March 2005, ACM.

[11] F. Buschmann and R. Meunier and H. Rohnert and P. Sommerlad and M. Stal, *A system of patterns, pattern oriented software architecture*, volume 1, Wiley & Sons, 1996.

[12] S. Nepal and J. Zic and S. Chen, *WSLA+: Web Service Level Agreement Language for Collaborations*, In Proceedings IEESCC, Nepal, Sept, 2008, pp. 485–488, IEEE.

[13] D. D. Lamanna and J. Skene and W. Emmerich, *SLAng: a language for service level agreements*, May 2003, IEEE.

[14] A. Andrieux and K. Czajkowski and A. Dan and K. Keahey and H. Ludwig and T. Nakata and J. Pruyne and J. Rofrano and S. Tuecke and M. Xu, *Web Services Agreement specification (ws-agreement)*, GRAAP Working Group of Open Grid Forum (OGF), September 7, 2006.

[15] E. Triantaphyllou and B. Shu and S.N. Sanchez and T.Ray, *Multicriteria Decision Making: An Operations Research Approache*, Enciclopedia of Eletrical and Elettronics Engeneering, 15, pp. 175-186, 1988

Improving the Performances of a Grid Infrastructure by means of Replica Selection Policies

Rosario Giunta[1], Fabrizio Messina[1], Giuseppe Pappalardo[1,2], Emiliano Tramontana[1,2]
[1]Dipartimento di Matematica e Informatica, Università di Catania, Italy
[2]Consorzio COMETA, Via Santa Sofia 64, I-95123 Catania
{giunta, messina, pappalardo, tramontana}@dmi.unict.it

***Abstract*—Current Grid middleware provides support for storing large data files and for handling a catalogue of replicas. Users wishing to access a file can arbitrarily choose one among available replicas. However, users are generally unaware of the state of Grid resources. As a result, their choice of the source, for the replica they need, cannot take into consideration critical factors like network latency, topology, congestion, etc.**

This paper describes a set of replica selection policies that are based on several parameters characterising the global state of a Grid infrastructure. The aim of our selection policies is to minimise the time that is needed to transfer the requested file from a source host, one among those holding a replica of the requested file, to the destination host. Essentially, the policies employed try to spread the load of the hosts serving a replica, as well as minimise the latency between source and destination hosts.

For these policies to be effective, data about the Grid infrastructure have to be known beforehand. Such data include the latency and bandwidth of communication links and the changing conditions of the workload of hosts serving files.

We have compared the transfer times recorded from the source hosts suggested by our selection policies, and found that the best source host depends on several factors including latency, workload and transfer protocol employed. In particular, we managed to partition the ranges where these factors vary in few intervals, identifying typical use scenarios. In each of these, at least one of our proposed selection policies has been found to perform satisfactorily.

***Index Terms*—Replica file, network latency, transfer protocols, performance.**

I. Introduction

GRID middlewares usually offer services to store and retrieve copies of data files. *Replica* files are created as a *master* file is copied from a host to others, in order to improve fault-tolerance and favour access by users from different locations. Grid middlewares usually provide tools to create, access and delete replica files. E.g., the Globus Toolkit [3] Grid middleware provides the Replica Location Service [4].

Such a conventional replica service is usually blindfolded to the changing state of Grid resources, such as the load on storage hosts, called Storage Elements (SE). As a result, it cannot provide any help in advising the user as to which SE might serve the replica with the lowest transfer time. In this paper, we propose two replica selection policies aimed at transparently providing the user with the best replica, given the latest states of the hosts holding the requested replica. Both our policies, called *fair-load* and *min-dist*, aim at minimising the replica transfer time experienced by the user. For this purpose, policy *fair-load* distributes the load among several hosts, whereas policy *min-dist* minimises the distance between the user and the served replica. Through extensive simulation, we gained confidence that our replica selection policies perform reasonably well in various,

significant contexts. A custom-made simulator was developed for this purpose.

Implementations of our replica selection policies must have access to a software layer maintaining fresh data about the state of SEs. For this purpose we devised a dedicated service, named LoadKeeper. Clients asking for a replica access this service *transparently*, using software *aspects* [6]. Thanks to *aspect-orientation*, existing clients need not be re-coded to benefit from the replica-location capability offered by LoadKeeper. It suffices to *interweave* suitable replica-selection *aspects* with the clients, without any need to change their original source code. Aspects intercept the API calls made by the client asking for a replica, and take control by applying the chosen policy, eventually providing the client with the location of a convenient replica.

This paper is organised as follows. Our replica selection policies, as well as a state-of-the art one, are described in Section II. The software layer allowing data to be collected is detailed in Section III. The custom-made simulator used to analyse and compare the noted policies is illustrated in Section IV. Grid resources that have been modelled for the purpose of simulations are described in Section V. Experimental results are reported in Section VI. We discuss some related work in Section VII, and finally draw some conclusions in Section VIII.

II. Policies for replica selection

In this section we detail the policies we propose for selecting one among a set of available replicas in a way leading to minimise response time. These policies use data on the current state of SEs on a Grid infrastructure. We also describe an existing replica selection policy, named *Rabinovich* after one of its authors. Later, we will compare the results provided by all these policies.

A. *Policy* fair-load

In a Grid system each host storing a replica can be contacted with an unpredictable frequency, causing an also unpredictable load on the host. Policy *fair-load* aims at distributing the load among hosts so as to avoid situations whereby a host is exceedingly busy serving replicas while others are idle. When such a situation occurs, the load on the host serving too many requests, with respect to its processing power, will cause a general slowdown in each request it is called upon to serve. Moreover, such an overloaded host is likely to stress, if not use up entirely, the bandwidth available over the network links it is attached to.

This policy makes sure that requests, which are intercepted by means of an aspect, will be redirected to SEs that are currently less loaded. For this it is only necessary to determine the load borne by hosts storing the requested replica and then refer the client to the less loaded host, i.e. the host having the smallest load. As the load factor for a host, we use the number of current requests.

B. *Policy* min-dist

Typically, a Grid system consists of hosts distributed amongst different locations, even spanning across different nations. In such a system, transfer times can be largely affected by the physical distance between hosts, because of the latency times introduced by network links.

Policy *min-dist* basically aims at choosing the host for serving a replica that is the nearest to the requesting host. However, to avoid overloading the nearest host when the Grid system has to serve many concurrent requests, this policy tries to compromise between distance and load. Always choosing the nearest host is inappropriate since such a host could become quickly overloaded, resulting in slowing down all the requests it will serve, despite it being near to requesters. This intuition has been confirmed by our experiments.

The definition of this policy is based on two factors that represent *distance* and *load* respectively. A score function is defined, for each host h holding the requested replica and the requesting application a, as

$$score(a, h) = \alpha \cdot dist(a, h) + \beta \cdot load(h)$$

where $load(h)$ represents the load on host h and $dist(a, h)$ represents the latency between the requesting application a and host h. Both $dist$ and $load$ have been normalised in the range [0, 1]. The parameters α and $\beta = 1 - \alpha$ are positive weighting factors used to tune the $score$ function; they are required to satisfy the constraints: $\alpha + \beta = 1$, $\alpha > 0$, $\beta > 0$.

C. *Policy* Rabinovich

In [8], Rabinovich et al. propose a strategy for web cache replication and selection. Their proposal is based on two values for every replica of every host: (i) a request counter (cnt), i.e. a number representing the times a replica file has been chosen on that host, and (ii) an affinity value (aff), i.e. a preference factor counting how many times the replica has been migrated to that host. In our implementation, the affinity value is always set to 1 because we do not migrate replicas, as we are only concerned with the selection problem.

When a request for a replica is made, the best replica is chosen comparing two values: (i) $cnt/2aff$, computed for the host closest to the application, and (ii) cnt/aff computed for the least requested host holding the replica. The host yielding the minimum value is used to serve the request. In this way a nice trade-off is obtained between performance and load-balancing. Indeed, the load will be reasonably spread over all hosts, as the closest host is chosen at most twice as much any other one.

III. Supporting software architecture

A typical Grid application accesses a logical file by passing the filename to an interface such as that exposed by the Globus Toolkit's Replica Location Service (RLS) API. This operation allows the application to obtain a list of all the hosts serving replicas of the desired file. From this list, the application can choose which host to contact to get the replica.

The list of possible sites holding a replica is managed by the middleware and reflects the current state of the Grid file system. Unfortunately, such catalogues only have *static* data, i.e. data determining which host holds which replica. To shorten transfer times, additional data are needed about the *dynamic* state of the Grid, i.e., for our purposes, the runtime load on SEs, and their distance from the application sites.

Such data are collected by service LoadKeeper, which continually receives updates about the current resource state from suitable aspects tied to clients. LoadKeeper mainly keeps track of the opening and closing operations on replicas held by SEs. Specifically, LoadKeeper maintains data about the replica access, i.e. the time-stamp for the opening and closing of any transfer, and the host where it is performed.

The client application is enhanced with the ReplicaMonitor aspect. This aspect monitors the running application and intercepts any use of the RLS API. When a replica file is opened, ReplicaMonitor contacts service LoadKeeper and sends the details of the operation. Therefore, each client will update LoadKeeper with fresh data that can be used to choose the best replica for each request.

The client is also interwoven with aspect ReplicaSelector, which, upon intercepting a request for a replica, will ask LoadKeeper the list of the best hosts according to the current Grid state. It will then pass back this list to the application. Hence the choice made by the application will be based on the dynamic state of the Grid and the specific selection policy implemented in LoadKeeper (among those introduced in Section II).

Thanks to aspect technology, components gathering and filtering data are separated from the clients and, being also independent of any client, are also easily reusable. Moreover, applications need not to be manually recoded to make use of such services, it is only necessary to (mechanically) interweave the replica-location related aspects with the original application.

IV. Simulator

To validate the above policies, we have built a custom simulator, based on the GridSim [5] libraries, which can be used to model several scenarios. Simulations have been launched in a real Grid environment, using the hosts of COMETA's Pi2S2 project. Thanks to such an environment, we were not limited to launch a single simulation at a time, but we could run many parallel simulations, gathering many more data in less time. The results obtained are commented in Section VI.

The described aspects have been implemented in the simulator and act as if in a real environment, communicating with service LoadKeeper. We have also measured the overhead introduced by the use of the aspects, by running the same simulations with and without the aspects (i.e., for the latter case, we have manually modified the client code to let it exploit service LoadKeeper). The aspects add an indirection level and a communication delay in contacting service LoadKeeper, but simulations have shown that, in the scenarios we are working in, this additional delay is unnoticeable compared with the average transfer time of a file.

For the simulations to be accurate we modelled the users of the Grid as acting according to the different patterns of behaviours described in Section IV-A.

The simulator, in addition to the obvious parameters such as file size, replication rate on hosts, etc., takes as input the topology of the simulated Grid and data characterising it, such as links bandwidth, latency, etc. The simulated networks are detailed in Section V.

A. Modelling users behaviours

Since each application[1] may behave differently from other applications, we expect to observe many different kinds of access patterns for file requests. E.g. an application can request a very large file, but spend a long time processing it before making another request. We aim at characterising several kinds of applications in our model, so as to investigate how our policies can be used in different working conditions of the Grid.

The behaviour of an application is modelled as the repetition of two basic operations: make a random transfer and then process the data. To model this behaviour, the main parameters we took care of are:

1) the average file size that the application requests; it can be set as constants *high* or *low* (expressed in MBs);
2) the frequency at which the application makes a file request; it can be set as *high* or *low* (expressed in GridSim simulation ticks);
3) the percentage of the total number of applications belonging to any possible category (a category is given by a combination of the two previous parameters).

By setting the values for these parameters, we are able to distinguish among several applications of different categories. A category identifies a set of applications that behave similarly, i.e., each makes requests with the same frequency, for files of similar size. Since we allowed for four possible combinations of the parameter values, we get four possible application categories. For each one, we provide the simulator with the number of applications to be assumed to belong to it.

By using such parameters we can simulate a great variety of scenarios, to test how the replica selection policies behave in different systems and network conditions. E.g., it is easy to model a scenario where some applications request small files and the others make frequent requests for large files. Thus, we can generate different usage patterns of the network bandwidth, in order to test our architecture in realistic scenarios.

V. Grid systems for the simulations

To test the proposed policies, we have gathered the necessary data of an existing large-scale Grid infrastructure and then reconstructed

[1] In the context of the simulator an application is an entity indistinguishable from a user.

sw 1	sw 2	4.781
sw 1	israel	39.436
sw 1	hung	5.546
sw 2	sw 5	13.661
sw 2	sw 3	1.631
sw 3	sw 4	0.656
sw 3	pd it	0.641
sw 4	ge it	5.973
sw 4	bo it	0.001
sw 5	infn	0.036
sw 5	inaf	0.423
sw 5	lns	0.034
sw 5	sw 6	0.004
sw 5	sw 7	0.013
sw 6	diit	0.035
sw 6	dmi	0.001
sw 7	pa	3.115
sw 7	me	5.374

TABLE I
LATENCY TIMES AMONG PAIRS OF HOST-SWITCH WITHIN THE GRID INFRASTRUCTURE (MS)

it in our simulator. Data collected are mainly static parameters such as the measured round-trip times between each pair of hosts. These data have been used to build a logical topology of the Grid infrastructure, i.e. a topology where each link and switch has been inserted according to the set of round-trip times measured and by using the algorithm described in [10]. Actually, the existing topology of the network connecting the hosts of the infrastructure would be much more complex, i.e. it would consist of a much greater number of links, switches, routers, etc. However, for all practical purposes, this simplification does not introduces errors, the round-trip times used being those measured on the real infrastructure.

Figure 1 shows the Grid infrastructure that we explored and then used for our simulations. Such an infrastructure spans across Europe and also reaches a site in Israel. The network links within the infrastructure possess very different characteristics, such as the round-trip times and bandwidth between sites. Table I shows the measures of latencies between each host and a switch and between pairs of switches, whereas the bandwidth of each link is depicted by the colours of the links.

The wideness of the network is reflected by the latencies of the communication links, determined by halving the round-trip times measured between each pair of hosts. Round-trip times range from a few microseconds, for resources that are actually within the same town (e.g. between sites *dmi* and *diit*), to several tens of milliseconds. It is worth noting that the longest distance between the sites is more than 2000 km, which explains the differences in the round-trip times reported.

The topology in Figure 1 is colour-coded to show the different bandwidths of the reconstructed communication links between the hosts. A black link represents a bandwidth of 155 Mbps, a red one stands for 2.5 Gbps and a blue one for 10 Gbps; all of these capacities are hand-picked to reflect the real topology of the Grid we are modelling. Figure 1 also shows the locations of software components and resources used for the simulations whose results are commented in Section VI. Specifically, we marked the replica hosts with Rn, the location for the Replica Catalog[2] with RC, the location for service LoadKeeper with LK and the location of the users making requests with U.

VI. EXPERIMENTAL RESULTS

Simulations have been performed for the Grid infrastructure described above, using both the TCP and the UDP protocol[3] and without any background traffic. To make the results comparable for the different scenarios we have set the file size for the replica to 100 MB and replicated every file in each host, so that there won't be any bottleneck when choosing from another host.

The behaviour for simulated users is set so as to make several requests for a replica, and in order to test the system under stress conditions, we have made simulations that request from 1 to 60 concurrent downloads simultaneously.

The replica selection policies we have compared are the ones described in Section II. We also compared the policies with a random

[2]It is the main catalog managed by the middleware.

[3]We are simulating a simple version of TCP that differs from UDP only by the use of ack messages.

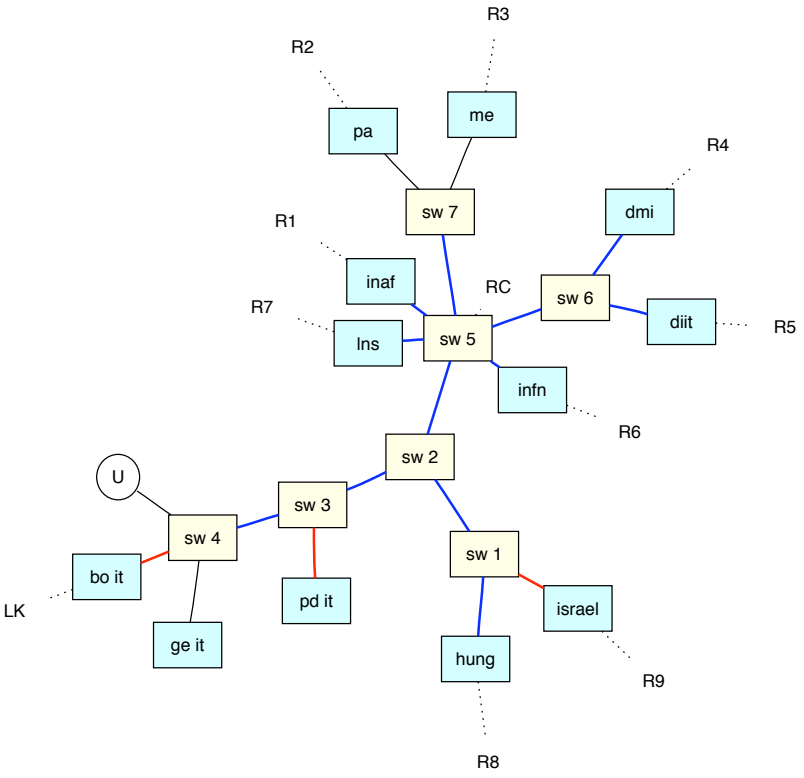


Fig. 1. Topology of the Grid infrastructure

replica choice to simulate the behaviour of an uninformed user.

Results of the simulations are shown in Figures 2 and 3. Each point on the figures shows the average transfer times when performing n concurrent transfers given the topology, the transport protocol and the policy used for the replica selection. The average is computed from a hundred simulations made with the same parameters.

The average transfer time in stress conditions grows faster when using TCP instead of UDP, as shown comparing the service times of Figure 3 with those of Figure 2.

Using UDP, our policies never exceed a 9 seconds transfer time for up to 60 concurrent replica requests. With the Rabinovich policy, the transfer times are almost always higher than our policies, in the presence of more than 10 simultaneous downloads.

Using TCP, in the same network, leads to transfer times that are higher for the Rabinovich policy in the presence of more than 35 simultaneous requests. However, we can clearly see that policy *min-dist* yields the lowest average transfer time from 20 simultaneous requests. We can say that for significant latency values[4] and highly loaded hosts, choosing according to *min-dist*, i.e. the nearest and not so loaded host, affords a significant gain. In some cases (e.g. 60 simultaneous requests), choosing according to *min-dist* gives a transfer time that is about 30% less than policy *Rabinovich*, and about 80% less than a random choice.

[4]When we simulated topologies with smaller latencies, *min-dist* performed not as well.

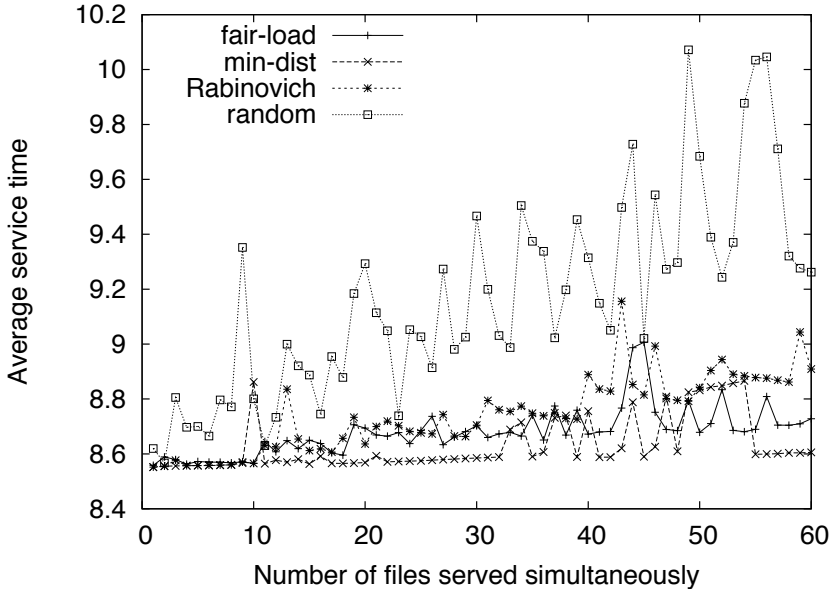


Fig. 2. Average service times for transferring a given number of files when using UDP

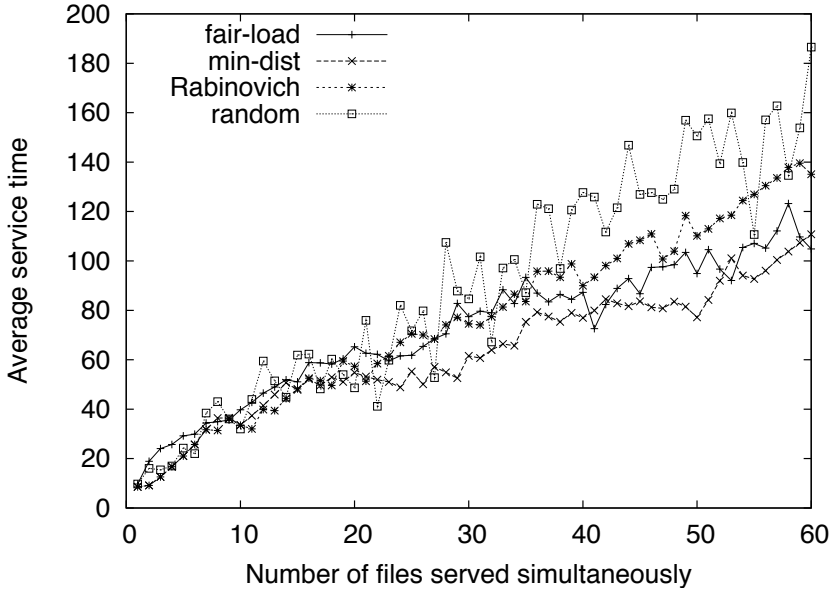


Fig. 3. Average service times for transferring a given number of files when using TCP

As for comparing the TCP- vs. UDP-based performances, we note that reliance on TCP seems to worsen the transfer times, which, using policy *min-dist*, may rise up to about ten times those obtained with UDP. This confirms that, in a network affected by high latencies, using TCP for downloading a large amount of data significantly affects transfer times. Indeed, especially when the nearest host cannot be used, since it is busy serving requests exceeding its capacity, the transfer times of TCP are in the average higher than with UDP. Note, e.g., that for less than 5 simultaneous transfers, the average transfer time with UDP is about half that with TCP, and rises to 10 times as much for 30 simultaneous transfers.

Figures also show that the transfer times for a random choice are nearly always longer than with the informed choice performed by the considered policies. Also, as expected, they tend to be very irregular and unpredictable.

Analysing the transfer times for the UDP transfer protocol (see Figure 2), it can be noted that the best choice, leading to shorter transfer times as the simultaneous request rate grows, is the *fair-load* policy, while the *min-dist* policy gives longer transfer times, albeit still better than the other two policies. Both our policies seem to grow linearly with the number of concurrent requests. The transfer times obtained with the Rabinovich policy also grow in a predictable way.

Using a confirmed protocol like TCP, instead, transfer times ensured by all policies are not too different. However, at the highest loads in terms of simultaneous request rates, it is quite clear that our *min-dist* and *fair-load* replica choice policies guarantee the minimum and second-best transfer times respectively.

VII. Related works

The file replication problem has been studied for several decades now and, while the results of such studies are certainly useful, they need to be specialised in order to be conveniently exploited in a Grid environment, which does exhibit some specificities with respect to a conventional scenario. E.g., the typical file stored on a Grid SE can easily have a size of some GBs, making it impractical to migrate its replicas from host to host; in fact, migration is likely to prove more useful and viable in a web-based environment, where cache files can be migrated rapidly due to their smaller sizes. Therefore, different approaches have been proposed to effectively tackle the replica selection problem, some of which we briefly describe in this section comparing them with our proposal.

The load balancing problem in a Grid environment is investigated in [9], which addresses the problem of migrating a job from the host running it to a less loaded one, given a set of buddy processors, each capable of estimating the load on others. The estimation is based on an information exchange that happens at fixed intervals, and a mathematical model of the distribution of a number of parameters such as the estimated arrival rate, the actual number of job departures, etc.

Two different approaches are proposed to manage the load in large and small scale Grids. In the former case, job migration is performed at fixed intervals, via the MELISA (Modified ELISA [1]) algorithm. For small-scale Grids, migration is considered every time a new job arrives, via the Load Balancing on Arrival algorithm. One key difference to our approach is that [9] is targeted at the job level, which only requires migrating small files (the jobs), and is therefore unsuitable (at least without significant changes) for our goal of managing replica files.

An approach especially devised for managing replica files in a Grid system is described in [2], where the authors use the Distributed Hash Tables for a Peer-to-Peer Replica Location Service. They make use of the Chord P2P protocol to build an overlay network to replicate the filename mappings without making changes to the original RLS. What the authors aim at is enhancing the fault-tolerance and scalability of the replica indexes; we do not address this problem as we are concerned with the choice of the best replica for a given file, so this approach

can be used complementarily to ours.

The system proposed in [7] deals with the query operator placement in a Distributed Stream Processing Network. In an overlay network, the authors periodically map a solution for the placement of the operators using bandwidth usage and distance as parameters for the relocation; the solution is found on a synthetic cost space and then mapped to the closest physical node. Using this approach for replica files would mean to change the function so as to employ it to select the best replica, searching the cost space based on different parameters. On the other hand, it would be impossible to exploit operator placement for our purposes in a straightforward way, as in replica management there is nothing that can be obviously seen as (the equivalent of) a query operator .

In [11] presents a predictive framework for the service time of a file transfer, based on mathematical predictors. The predictors employ the usage statistics recorded by a modified GridFTP service and can be of three different categories: average based, median based and auto-regressive model based. Unfortunately, as the authors themselves state, *"work with predictions was inconclusive"* because the accuracy of predictions varied significantly in their experiments, maybe due to the irregularities of the data.

VIII. Conclusions

In this paper, we have proposed to select the best replica file in a Grid environment by means of different suitable policies, instead of leaving the burden to the user. We support access to the selection policies by introducing a software layer that observes the action of the users and collects and sends appropriate data to a service that has a global vision of the usage of the SEs on the Grid. This service is then able to select the best replica as the nearest to the requesting user or that stored in the less loaded host.

Experimental results have been gathered by simulating a large-scale Grid infrastructure, which has been built according to a real network of Grid resources. Both UDP and TCP have been considered as transport protocols. Simulations have identified one of the proposed policies, which tries to compromise between load-balancing and distance, as that capable of ensuring the best transfer times in a significantly large number of different use scenarios.

IX. Acknowledgements

This work has been supported by project PI2S2, managed by Consorzio COMETA and co-funded by the Italian Ministry of University and Research (MIUR) within Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-06) (www.pi2s2.it, www.consorzio-cometa.it).

The authors wish to thank Emilio Mastriani, Costantino Pistagna and Luigi Toscano for their contribution.

References

[1] L. Anand, D. Ghose, and V. Mani. ELISA: An estimated load information scheduling algorithm for distributed computing systems. *Int'l J. Computers and Math. with Applications*, 37(8):57–85, Apr. 1999.

[2] M. Cai, A. Chervenak, and M. Frank. A peer-to-peer replica location service based on a distributed hash table. In *Proceedings of Supercomputing Conference*. ACM/IEEE, 2004.

[3] Globus Alliance. Home page and publications. *http://www.globus.org*.

[4] Globus Alliance. Replica location service. http://www.globus.org/grid_software/data/rls.php.

[5] Grid Computing and Distributed Systems (GRIDS) Laboratory. *GridSim: A Grid Simulation Toolkit for Resource Modelling and Application Scheduling for Parallel and Distributed Computing*, 2005. `http://www.gridbus.org/gridsim/`.

[6] G. Kiczales, J. Lamping, A. Mendhekar, C. Maeda, C. V. Lopes, J. M. Loingtier, and J. Irwin. Aspect-Oriented Programming. In *Proceedings of the European Conference on Object-Oriented Programming (ECOOP'97)*, volume 1241 of *LNCS*. Springer-Verlag, 1997.

[7] P. Pietzuch, J. Ledlie, J. Shneidman, M. Roussopoulos, M. Welsh, and M. Seltzer. Network-aware operator placement for stream-processing systems. In *Proceedings of International Conference on Data Engineering*. IEEE, 2006.

[8] M. Rabinovich, I. Rabinovich, R. Rajaraman, and A. Aggarwal. A dynamic object replication and migration protocol for an internet hosting service. In *Proceedings of International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 1999.

[9] R. Shah, B. Veeravalli, and M. Misra. On the design of adaptive and decentralized load balancing algorithms with load estimation for computational grid environments. *IEEE Trans. on Parallel and Distributed Systems*, 18(12), 2007.

[10] T. Shirai, H. Saito, and K. Taura. A fast topology inference: a building block for network-aware parallel processing. In *Proceedings of international symposium on high performance distributed computing (HPDC)*. ACM, 2007.

[11] S. Vazhkudai, J. M. Schopf, and I. T. Foster. Predicting the performance of wide area data transfers. In *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2002.

Measuring Performances of Globus Toolkit middleware Services

Rosario Giunta[1], Fabrizio Messina[1], Giuseppe Pappalardo[1,2], Emiliano Tramontana[1,2]
[1]Dipartimento di Matematica e Informatica, Università di Catania, Viale A.Doria 6, I-95125 Catania, Italy
[2]Consorzio COMETA, Via S.Sofia 64, I-95123 Catania, Italy
{giunta, messina, pappalardo, tramontana}@dmi.unict.it

***Abstract*—Generally, users working on a Grid infrastructure have a keen interest in knowing (tight) upper bounds for execution times of useful Grid services and their own submitted jobs. However, in fact the execution environments and times in a Grid infrastructure may widely vary in practice because of several and diverse factors, including: the unpredictable delays on network links, peaks of service requests, the relatively unpredictable outcomes of resource negotiation, changes in services and resources availability due to partial failure, etc.**

It therefore appears that, for successful Grid operation, a crucial ability is that to analyse and accordingly tune the amount of resources needed to smoothly execute requests arising from running Grid services and user-submitted jobs. In order to make such a tuning possible and effective, an essential condition is the availability of a proper characterisation of the execution times of the services provided by the Grid middleware itself.

This paper focuses on those Grid services whose execution times can vary substantially, analysing what factors determine or affect this variability. We present a set of measures that have been carried out for Grid services handling job submission, on a real infrastructure, as usage conditions vary. This knowledge is very useful in order to detect the occurrence of anomalous states in Grid operation, and to timely start appropriate recovery actions in reaction.

***Index Terms*—Globus Toolkit, PBS, performance, job execution.**

I. Introduction

GLOBUS Toolkit (GT) is a widespread middleware that provides several services supporting the users who perform activities over a widely distributed set of resources. GT includes services for job execution, file transfer, resource discovery, user authorisation, etc. [6].

Similarly to what happens in other middlewares, the execution times of GT services are often unpredictable mainly because of potential delays on network links, peaks of requests for the same service, variable amount of resource negotiation, changes in services and resources availability due to partial failure, etc. On the other hand, the execution time of submitted jobs greatly varies because of the said variability affecting GT services, for the different requirements of jobs, and due to the changing availability of resources used to serve concurrent requests [7].

A monitoring architecture for measuring data related to the execution of jobs has been proposed in [7]. Job execution is monitored by connecting sensors to the job, then all the measures on hosts running jobs are passed to a higher-level component that transforms and aggregates data coming from different jobs.

An analysis of the behaviour of gLite middleware [4], which is a variation of GT having similar services and based on GT version 2.4.3, has been performed using Generalized Stochastic Petri Nets [1]. Such an analysis aims at gaining knowledge on what to expect for several usage scenarios of a set of Grid resources. Indeed, such a work makes some assumptions on the probabilistic distribution of the execution times

for several Grid services, such as e.g. time for the data to move from a Storage Element (SE) to a CPU. Moreover, it considers negligible the time taken for sending messages between services within the same cluster.

Each of the said contributions is concerned with the performances of user jobs, executing on top of middleware services, however not with the execution times of standard Grid services themselves, which actually affect the responsiveness of the Grid infrastructure.

This paper reports the response times that have been measured for the service handling job submissions, provided by GT, under specific settings, i.e. measures have been performed by running services on a dedicated set of hardware resources when varying the rate of homogeneous requests. The measures we provide here are real response times of the GT Grid middleware services and as such they could be e.g. used to: confirm the working scenario illustrated in [1], assess whether typical Grid resources are capable of providing the response time expected by users, tune the performance of a Grid infrastructure, etc.

Recent proposals aim at using Grid middleware in an environment where users may need to interact with applications to provide inputs and gather results in a more responsive way, i.e. users expect a quick reply from middleware and applications [8]. Our study shows the time scale for executing some services on a Grid and hence characterises in which conditions such operations can be expected as responsive.

This paper is organised as follows. Section II describes a typical Grid infrastructure. Section III illustrates the measures we are interested in. Section IV details how selected GT services have been modified in order to collect data. Section V reports the scenarios for taking measures, the actual data obtained and an analysis of such measures. Finally, Section VI draws some conclusions.

II. Deployment of GT Services

Typically, a Grid site consists of several Worker Nodes (WNs) providing the CPUs where jobs are going to run; a Computing Element (CE) handles one or more queues for the requests of job execution and feeds a WN with a job. Thus, a CE shields WNs of direct access from users and services external to the said CE. Additionally, a Grid site has one or more Storage Elements (SEs) supporting the storage and retrival of a large amount of data.

GT provides services for handling job execution, file transfer, resource discovery, authorisation, etc., each of these services consists of several components, some of which are only optional, thus not necessarily available on a given installation. From version 4 of GT, each service is provided as a command line and as Java, C, Python APIs.

The component that within GT provides services for submitting, monitoring and cancel jobs is called GRAM (Grid Resource Allocation & Management) [3]. GRAM relies on batch schedulers for local hosts, such as Condor, PBS or LSF, that handle job submission to WNs.

According to GT installation options [6], GRAM is typically deployed on a CE together with a local *resource manager*, such as PBS. This is responsible for handling job submissions often using several priority queues and for extracting a job from the appropriate queue when a WN becomes available. A server part of the resource manager PBS is deployed on each available WN to handle the reception of a job and for other communications. Figure 1 shows the main components of GT and the hardware resources where services of GT and the underlying support are typically deployed.

For obtaining a platform where measures of performances could be performed, the ensemble of services handling job submissions has been deployed on three hosts $host$ 1, $host$ 2, and $host$ 3 (see Figure 1) as follows. Machine $host$ 1 holds the GT components allowing the user to be authenticated and to submit jobs. E.g. $host$ 1 runs a User Interface (UI) and a SimpleCA, i.e. a Certification Authority, in order to create and sign certificates, hence allowing users to perform tests. Machine $host$ 2 holds the GT components providing: data man-

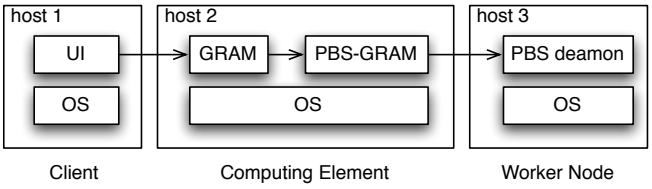


Fig. 1. Typical Grid infrastructure for executing jobs

agement (i.e. GridFTP), execution management (i.e. GRAM), and common runtime (i.e. Java/C WS Core and XIO). Moreover, *host* 2 holds PBS-GRAM, i.e. a component able to pass requests for job submissions received by GRAM to PBS scheduler[1]. Finally, *host* 3 holds PBS daemon, i.e. a component capable to receive jobs and execute these on the same host.

III. Measures Performed on GT Services

This section provides details of the response times that have been measured for the GT services handling job submissions.

GT version 4 handles job submission by means of command `globusrun-ws` and this takes as input a set of switches and parameters to characterise the type of submission and the job to be executed.

Once a user performs a request for job execution, the reaction of GT for handling such a request is organised so that several transitions between states will occur. The state of a job whose execution request has been performed is called *external state*, i.e. a state that is visible to the requesting clients, and it can be one among: *Unsubmitted*, *StageIn*, *Pending*, *Active*, *Suspended*, *StageOut*, *CleanUp*, *Done*, *Failed*.

The first set of measures we want to achieve are fine-grained measurements of the performances perceived by the user, i.e. at the client-side, of GT services that handle a request for job execution. Hence, we are interested in having the time intervals spent in each of the said external states for submitted jobs. These measures are performed when varying the number of simultaneous jobs submitted. The said fine-granularity measures provide support for understanding the working within GT services handling job submission. Since no timestamp is associated when, logically, a job is moved from a state to another, in order to measure the time frames elapsed in each of the different external states for the job, we had to instrument the C source code of `globusrun-ws` (see Section IV for relevant details).

The following gives an overview of the operations performed by GT in each of the external states for which we have collected measures (see Section V for the data).

- *Unsubmitted*, in this state the execution, and the time measured, is due to the operations performed on CE middleware, such as user authentication, authorisation, job data structures creations, etc.
- *Pending*, when in this state, the time measured is the waiting time that a job has spent in the local (PBS) queue after being passed to the local (PBS) scheduler.
- *Active*, when in this state the job is actually running on a WN, hence the corresponding measured time is the job execution time.
- *CleanUp*, when a job is in this state, after its completion, the execution time is due to operations that are performed to remove data related with the job and not used anymore by the middleware.

The second set of measures we are interested in are the executions times of GT services for handling job submissions, grouped according to relevant internal steps that are being performed by the middleware at the server-

[1]We have deployed Torque, that is the 'open' version of PBS system [2].

side. We have monitored the inner workings of GT middleware and hence measured the relevant time frames by analysing the standard logs of service ManagedJobFactoryService, which is executed on the target host, i.e. a WN, by `globusrun-ws`. Hence, for this second set of measures no instrumentation of GT code was needed. According to the log files provided by service ManagedJobFactoryService, we could be able to identify four main groups of operations, i.e.: *Authorisation* comprising several steps for the authentication of requests; *Job create* for the creation of job id; *Job submission* for transferring the job to a WN and executing it; and *Job post-processing* for freeing resources related to the job. As for the first set of measures, for this second set we also have requested to execute an increasing number of times the same job.

The groups of operations, whose measured execution times are given in Section V, are described in the following.

- *Authorisation*, represents operations performed, and hence the corresponding time measured, for user authentication, user authorisation and for the creation of the job data structures.
- *Job create*, groups the operations, and the time spent, for preparing the submission of the job to the PBS scheduler.
- *Job submission*, characterises, and provide the measured time for, the job waiting on the PBS scheduler queue and the job executing on a CPU.
- *Job post-processing*, represents operations, and corresponding time, for the cleanup phase after the job has been completed.

IV. Instrumenting Service `globusrun-ws`

Thanks to GT modularity, we could be able to modify only a small part of its source code for our instrumentation, and then build and install only the module affected by changes.

The instrumentation of `globusrun-ws` consisted in inserting into the source code of function `globus_l_state_change()`, within file *globus_i_monitor.c* (which is part of `globusrun-ws`), an invocation to our custom built function `client_job_logger()` that records the current time and the transition into the new state. The standard version of function `globus_l_state_change()`, i.e. as we have found in the implementation provided by GT, receives a parameter identifying the new state and is called every time the external state of the job changes. Thus, the said instrumentation suffices to capture the instant of change and the current state for the job.

After having modified the source code of `globusrun-ws` we have built and installed, on top of our installation of GT, only the instrumented version of this module. In order to enable the selective installation, option `enable-maintainer-mode` has been passed to command `configure`, just before launching command `make` [5].

V. Execution Scenarios and Measures

We have measured the performances for handling job submission and executing jobs for three different scenarios. These will help highlighting the performances of GT services.

For each scenario, we ask to execute an increasing number of times the same job that resizes a JPG image of about 2 MB[2]. The job performs the transformation of the image three times, without any stageIn/stageOut operation.

For the execution testbed, we have used an eight-core Intel Xeon @ 2.50 GHz having in overall 8 GB RAM, and each of the (virtual) hosts in Figure 1 has been given 1 GB RAM. Each virtual host runs Debian Gnu Linux with kernel 2.6.26 (i686), $host$ 1 and $host$ 2 run GT version 4.2.1, which was installed from the source code, $host$ 2 and $host$ 3 run Torque (PBS) version 2.3.6. The networking among virtual hosts is handled by Virtual LAN (Qemu).

In our execution scenarios, we have measured performances when having a limited number of resources compared with the rate of requests.

[2]All requests of executions are performed in rapid succession, i.e. all within the first second of execution.

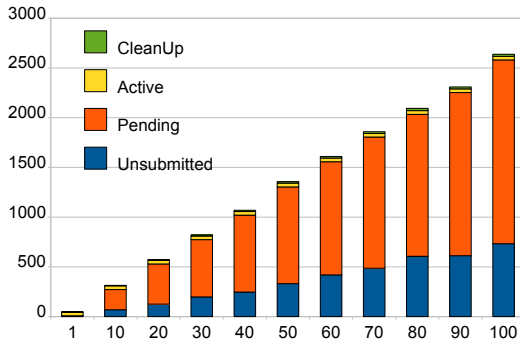


Fig. 2. Measured average times spent in each external state of the job (sec) vs number of simultaneous requests of job execution

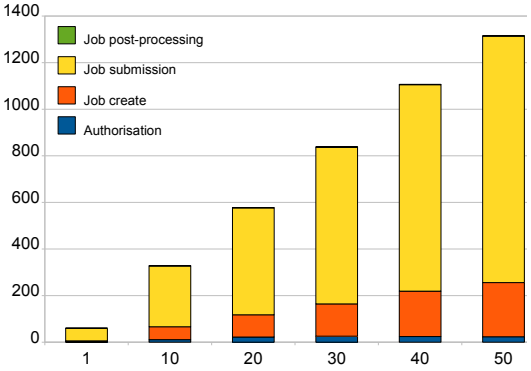


Fig. 3. Measured average execution times for internal GT operations (sec) vs number of simultaneous requests of job execution

I.e. the rate of requests for executing jobs is higher than the rate of jobs terminating their execution, hence the job waiting time increases.

A. Job submissions using GT

In the first scenario, named user-globus, a user submits jobs to be executed from *host* 1 (see Figure 1) by launching from the shell the command `globusrun-ws` and passing to this a parameter specifying as a URL the target machine, i.e. *host* 3 and service ManagedJobFactoryService. Actually, this service sends the job to the PBS scheduler, which in turn enqueues the job for execution on *host* 3.

Figure 2 shows the average execution times, while varying the number of jobs submitted, for each of the external states of the job (see Section III), perceived at the client-side. The bars show the total times as cumulative times, based on state transitions described in Section III.

Note that both the time frames spent in state *Unsubmitted* and in *Pending* grow with the number of concurrent requests for executing jobs. Whereas the time frames spent in the other two states, *Active* and *CleanUp*, are almost unchanged.

While we expected that the time spent in *Pending*, due to the need of waiting for an idle CPU, would increase with the number of simultaneous requests, the growth of time spent in *Unsubmitted* is unexpected (also it is difficult to explain) and seems independent of whether the job will be actually executing soon after the corresponding request or will be waiting for other jobs to finish before starting executing.

Figure 3 shows the average execution times, while varying the number of jobs submitted, for each of the main inner operation-groups performed by GT to handle job execution (see Section III), measured at the server-side.

Note that, while varying the number of simultaneous requests, the time frames measured for operation-groups *Authorisation* and *Job post-processing* are almost unchanged, while for operation-groups *Job create* and *Job submission* the time frames measured grow with the increasing number of simultaneous requests.

In a similar way to what happens when analysing the external states of a job, it is unexpected that the time spent for *Job create* grows according to the number of requests.

B. Job submissions using PBS

In the second scenario, named user-pbs, a user submits jobs from *host* 2 by invoking from the shell command `qsub`, which is provided by the PBS Torque scheduler. Command `qsub` takes as input a shell script specifying the commands to be executed, it is not as powerful nor configurable as `globusrun-ws`, however it suffices for our tests. In this scenario we ask

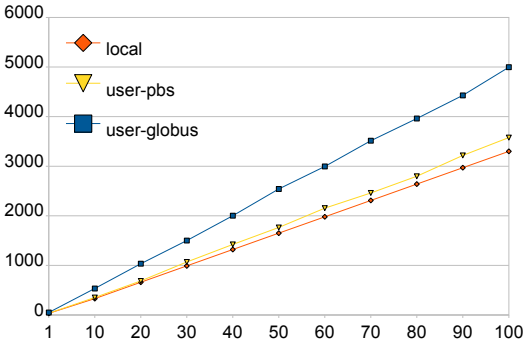


Fig. 4. Measured total execution times (sec) vs number of simultaneous requests of job execution

for executing the same jobs as scenario user-globus without using GT, thus by comparison we recognise the overhead given by GT.

C. Job submissions using an ideal machine

In the third scenario, named local, the user launches the execution of jobs using the OS shell on *host* 3. This scenario, skips both GT and PBS layers and provides the lowest possible execution times, since only the OS scheduler is actually handling the requests. Of course, this scenario is not realistic, since we need to handle submissions to several host, however it is used only for the sake of comparing previous measures with an ideal scenario.

D. Comparing execution times

Figure 4 shows the total execution times for executing a variable number of jobs whose requests for execution has been performed simultaneously. As expected each layer of the distributed system introduces an execution delay over the other layer. Only for the sake of comparing performances, we show that the lowest possible execution time is given by scenario local. The overhead given by scenario user-pbs amounts to about 10% over local and it uniformly affects execution times when varying the number of simultaneous requests.

The overhead given by GT (see scenario user-globus) amount to about 35% over user-pbs.

VI. Conclusions

This paper has measured the times related with the execution of GT services which handle the submissions of jobs. Two set of measures have been performed: at the client side, i.e. the side sending requests of executions, by intercepting the changes between external states for the jobs; and at the server side, i.e. at the CE, by analysing standard logs. Both set of measures have shown that the time needed by GT to handle requests for execution increases with the number of simultaneous requests. Such a time seems independent of the time that the job has to spend waiting for execution, when the availability of resources is limited.

The total time needed for executing a given number of jobs has been measured for three different scenarios, i.e. when using GT in combination with PBS, when using PBS only, and on an ideal machine. Such a measured time provides means for comparing the overhead imposed by each layer of a Grid infrastructure.

VII. Acknowledgements

This work has been supported by project PI2S2, managed by Consorzio COMETA and co-funded by the Italian Ministry of University and Research (MIUR) within Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-06) (www.pi2s2.it, www.consorzio-cometa.it).

References

[1] D. Bruneo, M. Scarpa, and A. Puliafito. A GSPN model to analyze performance parameters in gLite Grids. In *Proceedings of International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. IEEE, 2008.

[2] Cluster Resources. Torque resource manager. *http://www.globus.org*.

[3] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke. A Resource Management Architecture for Metacomputing Systems. In *Proceedings of 4th International Workshop on Job Scheduling Strategies for Parallel Processing*, volume 1459 of *LNCS*, pages 62–82. Springer-Verlag, 1998.

[4] gLite: Lightweight Middleware for Grid Computing. Home page. *http://glite.web.cern.ch/glite*.

[5] Globus Alliance. Globsu developer guide. *http://www.globus.org/toolkit/docs/4.2/4.2.1/developer/*.

[6] Globus Alliance. Home page and publications. *http://www.globus.org*.
[7] Z. Nemeth, G. Gombas, and Z. Balaton. Performance Evaluation on Grids: Directions, Issues and Open Problems. In *Proceedings of Parallel, Distributed and Network-Based Processing*. IEEE, 2004.
[8] G. Novelli, G. Pappalardo, C. Santoro, and E. Tramontana. A grid-based infrastructure to support multimedia content distribution. In *Proceedings of UPGRADE-CN Workshop of HPDC*. ACM, 2007.

The ASTRA Project Achievements: The Reconstructed Greek Epigonion

Salvatore Avanzo[1], Francesco De Mattia[3],
Giuseppe La Rocca[4], Mariapaola Sorrentino[2], Domenico Vicinanza[2,5]

[1]*University of Salerno – Department of Mathematics and Computer Science – SoundLab*
salvatore.avanzo@gmail.com
[2]*Conservatory of Music of Salerno, Via S. de Renzi, 62, I-84125 Salerno, Italy*
mariapaola@astraproject.com
[3]*Conservatory of Music of Parma, Via del Conservatorio, 27/A, I-43100 Parma, Italy*
francesco.demattia@conservatorio.pr.it
[4]*Istituto Nazionale di Fisica Nucleare, Sezione di Catania, Via S. Sofia 64, I-95123 Catania, Italy, giuseppe.larocca@ct.infn.it*
[5]*DANTE, City House, 126-130 Hills Road, Cambridge CB2 IPQ, UK*
domenico.vicinanza@dante.org

Abstract— ASTRA (Ancient instruments Sound/Timbre Reconstruction Application)[1] is a project coordinated at Conservatory of Music of Parma which brings history to life. Ancient musical instruments can now be heard for the first time in hundreds of years, thanks to the successful synergy between art/humanities and science. The Epigonion, an instruments of the past has been digitally recreated using an advanced EGEE [2] middleware (with the GENIUS [3][4] interface) and GÉANT2 [5] and EUMEDCONNECT [6] research networks to link high capacity computers together, sharing information to enable the computer-intensive modelling of musical sounds. This paper summarizes the most recent outcomes of the project underlining the GRID aspect of the computation and the results.

***Index Te1ms*— Grid Computing, Physical Modelling Technique, the ASTRA project.**

I. INTRODUCTION

ASTRA projects aims to reconstruct the sound or timbre of ancient instruments (not existing anymore) using archaeological data as fragments from excavations, written descriptions, pictures.

The technique used is the physical modeling synthesis, a complex digital audio rendering technique which allows modeling the time-domain physics of the instrument.

In other words the basic idea is to recreate a model of the musical instrument and produce the sound by simulating its behavior as a mechanical system. The application would produce one or more sounds corresponding to different configurations of the instrument (i.e. the different notes).

A. Modelling ancient instruments

To model ancient instruments not existing anymore we used a technique called physical modeling[7]. The physical modelling sound synthesis technique aims in fact to use a set of equations and algorithms that describe the physical materials used in the ancient instruments to generate physical source of sound. In other words, the basic idea is to recreate a model of the musical instrument and produce the sound by simulating its behavior as a mechanical system. This would lead to a

natural synthesized sound, correctly reproducing the instrument mechanics.

For example, to model the sound of a drum, a formula for how striking the drumhead injects energy into a two dimensional membrane would be created. Then, the properties of the membrane (mass density, stiffness, and so on), its coupling with the resonance of the cylindrical body of the drum, and the conditions at its boundaries (a rigid termination to the drum's body) would describe its movement over time and how it would sound.

Similar modelling can be carried out in instruments such as a violin, though replicating the slip-stick behavior of the bow against the string, the width of the bow, the resonance and damping behavior of the strings, the transfer of string vibrations through the bridge, and finally, the resonance of the soundboard in response to those vibrations.

We can say then that an instrument model within ASTRA, consists of laws of physics that govern the sound production and evolution, and will typically have several parameters, some of which are constants that describe the physical materials and dimensions of the instrument, while others are time-dependent functions that describe the player's interaction with it, such as plucking a string, or covering tone holes.

In the ASTRA project we used archaeological data as input (for example fragments from excavations, written descriptions, pictures on ancient urns), so that the complex digital audio rendering technique was tuned to models the sound of the particular instrument under study. The result was then a virtual model of the ancient instrument, described as an algorithm which can be converted into a computer program to finally compute the waveform.

B. Computational aspects of physical modeling.

Although physical modeling is not a new concept in acoustics and synthesis, having been implemented using wave equations created by *Hiller* and *Ruiz* in 1971, its practical use has only just become feasible. Physical modelling is a very computing-intensive technique, as the complex models of the musical instruments are solved by integrating numerical coupled differential equations. For example, to correctly reproduce a sound lasting for 30 seconds takes more than four hours of CPU time. Figure 1. shows how the computation time grow up depending of the simulation time.

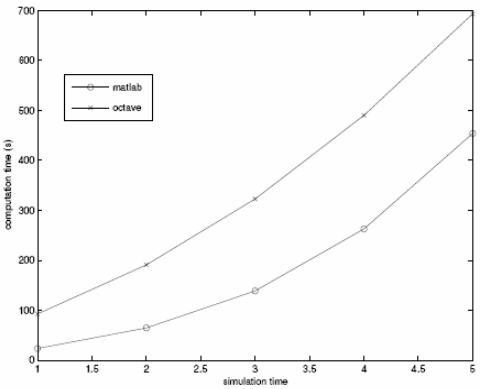


Fig 1 – Computation time vs simulation time

To finally reach the computational power needed by the project and speed up the sound reconstruction, the ASTRA project moved from a local computer cluster towards distributed computing resources. First tests have been successfully conducted within the GILDA facility[8], which bring together 20 sites (not all on-line at the same time) in 3 continents, and later on the EUMEDGRID [9] Grid Computing Infrastructurewhich link computing resources across the Mediterranean region at up to 2.5 Gbps through the GÉANT2 [5] and EUMEDCONNECT [6] research networks.

The extremely high reliability of the GÉANT2 network (see Fig 2), its performances in terms of number of institutions connected and throughput, made the ASTRA project possible.

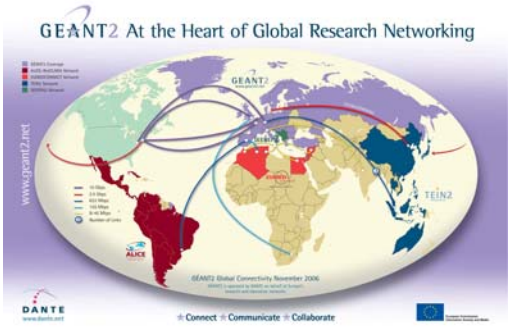


Fig 2 – The GEANT2 network connect 34 countries through 30 national research and education networks (NRENs), using multiple 10Gbps wavelengths

C. Running ASTRA in Grid.

In this section we describe the set up of the distributed environment that allows non expert users to exploit the huge computation resources of a grid infrastructure to reproduce the sound of ancient instruments on a large-scale. In order to achieve this strategic and ambitious goal some tasks have been pursued by the *Conservatory of Music of Salerno*, the *Conservatory of Music of Parma* and the Italian INFN. First and foremost the distributed environment of the GILDA t-Infrastructure first and EUMEDGRID later have been successfully configured installing, on the dedicated software experiment directories, all the external libraries requested during the computation by ASTRA. The installation of the new software for ASTRA has been published with a specific software tag in the top BDII for the given VO. The installation and the publishing of the new software tag for ASTRA has been conducted by the Software Manager in order to make easy future updates of the software on the resources. Another important task pursed by all the partners involved in the projects to foster the exploitation of the Grid paradigm from musicians, archaeologists and historians was to create a web-server interface to go through the initial difficulty in using the Grid middleware by users who are not familiar with it. There is a sort of *"scientific gap"*, represented by the difficult of complex computing protocols and rather complex Command Line Interface, that modern scientists have to learn before to access grids and gain benefits. To overcome this problem a web-server interface has been designed into the GENIUS Grid Portal[3][4]. This computing web portal, powered by EnginFrame provides to the end-user a transparent and abstraction layer to access the distributed services and resources of a typical grid infrastructure.

D. The GENIUS Architecture and EnginFrame

EnginFrame is a Web-based technology, developed by the Italian company Nice[11] S.r.l., that enables the access and the exploitation of Grid-enabled applications and infrastructure. It allows users, grouped within of a VO, to exploit the whole power of a grid infrastructure hiding all its complexity to the end-users and providing an abstraction layer to the grid technologies. *EnginFrame* is also well known in the Grid research world for being the technology which *GENIUS (Grid Enable web eNvironment for site Independent User job Submission*), the official Grid portal of the European project EGEE is based on. Due to the modularity and flexibility of EnginFrame, the Genius Portal can be easily customized to interact with several Grid component and/or VO allowing scientists to access, execute and monitor their own applications only by using a simple web browser. In this way the complexity of the underlying Grid infrastructure is hidden to users. The features exposed by EnginFrame and GENIUS portal contributed to speed up the sound reconstruction of ancient instruments and to disseminate the achievements during demo sessions organized by ASTRA. Fig. 3 shows the web-server interface designed in GENIUS to run ASTRA in grid.

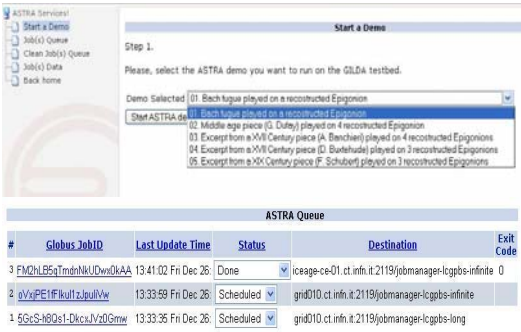


Fig 3 – ASTRA and the GENIUS Grid Portal

In the top figure the user selects the demo he/she wants to run on the Grid. In the bottom figure the monitoring of the sound reconstruction process is shown. Once the job' status is Done and exit code is 0, user can download the results produced by ASTRA and listening the sound reconstructed with the Grid. In Fig. 4 the distributed environment set up for ASTRA is shown:

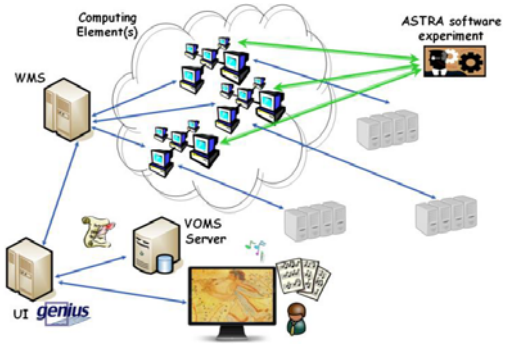


Fig. 4 - The application workflow set up for ASTRA

E. ASTRA results

In this section the sounds of some ancient instrument reconstructed by ASTRA are shown. Since September 2008, the Epigonion (an harp-like musical instrument from Ancient Greece) sound is finally available. To achieve this we used the advanced GÉANT2 [5] and EUMEDCONNECT [6] research networks to link high capacity computers together, sharing information to enable the computer-intensive modelling of musical sounds.

Knowledge of the Epigonion musical instrument, dating back from the Ancient Greek era, was based on archaeological findings, historical pictures and literature. It was a wooden string instrument that musicians have likened the sound to something similar to a modern harp or a harpsichord. We have compiled the sounds of four Epigonion instruments to recreate a medieval musical piece, making this the first time that these instruments have been heard performing together.

Another important result was a concert held in Naples, Italy on 14 December 2008. It was the first time ever that an instrument of the past reconstructed via compute-intensive modelling performed alongside real instruments such as violins and flutes as well as voices. The Psalm "*Laetatus sum*" by Czech composer *Jan Dismas Zelenka* was performed by the Sonora Network Ensemble enriched with the digitally reconstructed sounds of the ASTRA Epigonion.
Promoted as a concert produced in the 'listening laboratory' the performance aimed at paving the way for innovative musical research which enables a wide range of scientists, such as archaeologists, network engineers and computer experts to collaborate with artists to revive lost sounds of the past in real concerts (a demo of the concert is available here[12]). Other sound reconstruction examples are :

- Middle age piece (*G. Dufay*) played on 4 reconstructed Epigonions, wooden resonant structure, plucked and hammered strings[13];
- Excerpt from a XVII Century piece (C. Monteverdi) played on 4 reconstructed Epigonions, wooden resonant structure, plucked and hammered strings [14];
- Bach fugue played on a reconstructed series of monochords, wooden

resonant structure, plucked strings [15].

II. Conclusion

The benefits of this project are far reaching because ASTRA makes it possible to recreate the sound of instruments that previously would have been either too expensive or too difficult to manufacture by hand.
The world premiere concert held in Naples clearly showcased the immense potential for creativity when two worlds come together: art/music on the one side and science/technology on the other. With their huge computing power essential for the sound modelling, high-speed networks such as GÉANT2 [5] and EUMEDCONNECT [6] act as the facilitator for this creative union.

The project is running since 2006 thanks to the GEANT2 backbone, allowing researchers, musicians and historians to collaborate, communicate and share experiences on lost instruments and sounds ASTRA brings again to life. GENIUS Grid Portal is the official portal of GILDA[8], a complete t-Infrastructure for grid dissemination and training set up and managed by the INFN in the context of the EGEE Projects [2]. For any information or enquiry info@astraproject.org.

Acknowledgments

We gratefully acknowledge all the people who supported this work contributing with ideas, requirements and feedback.

References

[1] The ASTRA Project web site – www.astraproject.org
[2] The EGEE Project – public.eu-egee.org/
[3] Andronico G, Barbera R, Falzone A, Lo Re G, Pulvirenti A, Rodolico A **– "GENIUS: a web portal for the grid" –** Nucl. Instrument and Methods in Phy. Res. A 2003. Visit also the official GENIUS web site: https://genius.ct.infn.it/
[4] Barbera R, Falzone A, Ardizzone V, Scardaci D – "**The GENIUS Grid Portal : Its Architecture, Improvements of Features ,and New Implementations about Authentication and Authorization"** – Enabling Technologies: Infrastructure for Collaborative Enterprises, 2007. WETICE 2007. 16th IEEE International Workshops on..
[5] The GÉANT2 – www.geant2.net/
[6] The EUMEDCONNECT Project www.eumedconnect.org
[7] Physical Modelling (sound synthesis technique): en.wikipedia.org/wiki/Physical_modelling_synthesis
[8] Andronico G, Ardizzone V, Barbera R, Catania R, Falzone A, Giorgio E, La Rocca G, Monforte S, Pappalardo M, Passaro G, Platania G, **– GILDA: The Grid INFN Virtual Laboratory for Dissemination Activities.** Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005 **–** Visit also the official GILDA web site : https://gilda.ct.infn.it/
[9] The EUMEDGRID Project – www.eumedgrid.org
[10] EnginFrame – www.enginframe.com
[11] NICE Srl – www.nice-italy.com
[12] ASTRA Concert Demo – www.astraproject.org/files/concertdemo.mov
[13] www.astraproject.org/examples/dufay.mp3
[14] www.astraproject.org/examples/monteverdi.mp3
[15] www.astraproject.org/bachfugue.wav

Visualization of 3D Laser Scanner Data from an Archaeological Site Using OpenSceneGraph in ENEA-GRID Environment

D'Abate[1], M.C.Baracca[2], R.Ciavarella[2], G.Furini[2], S.Migliori[2] and S.Pierattini[2]

[1]*Università degli Studi "G. D'Annunzio", Chieti-Pescara, Italy, d.abate@unidav.it*
[2]*ENEA, Roma, Italy,*
{cristina.baracca, roberto.ciavarella, graziano.furini, silvio.migliori, samuele.pierattini}@enea.it

Abstract— **One of the main issue to be faced in archaeology is monitoring the conditions of a given site during a certain period of time in order to document progresses on the job. This issue can be resolved periodically performing 3D laser scanning of the site. In this way we can keep the whole area interested by the archaeological dig under control. Each 3D scanning yields a file containing millions of coordinates, each of which represents a point of the archaeological site. A color label in the format RGB is associated to each point. At the end we usually obtain a cloud of some millions of points. In order to meet the needs of archaeologists who work "on the ground", those data should be managed by applications capable to display a large amount of data in a relatively short time. It allows an operator to see the cloud points "nearly in real-time" while he is scanning an archaeological site; it also permits to carry out operations on cloud points such as measures between points, pull plants, sections, etc.**

***Index Terms*— ENEA, Laser Scanner Tecnology, 3D Model, Point Cloud, Grid Computing, OpenSceneGraph, Archaeology.**

I. INTRODUCTION

One of the main issues to face in archaeology is the monitoring of a site during a period of time, in order to document the progress of the activity. This issue can be solved via periodic 3D laser scanning of the site. In this way the whole area interested by archaeological digs can be kept under control.

Each file obtained via 3D scanning contains millions of x, y, z coordinates, each representing one point of the scanned archaeological site. A color label in the RGB format is associated to each point.

Usually, the final result of the process is a set of points, which may be displayed using the OpenSceneGraph [1] libraries and ENEA GRID environment [2].

The 3D model obtained by laser scanning is an objective database which allows to extract information about the site morphology, geometric structure and materials.

If the database contains a sufficiently high number of points, significant degradation phenomena can even be evaluated. Therefore, during post-processing activity on the 3D database, it is possible to retrieve plant, section and 2D elevation views.

II. THREE-DIMENSIONAL SCANNING OF AN ARCHAEOLOGICAL SITE

A. ENEA Laser Scanner

ENEA has a Time Of Flight (TOF) 3D Laser Scanner (Leica Geosystem, HDS 3000) [3].
HDS 3000 is a high speed and high precision laser scanner combining different features suitable for a wide range of applications.
The maximum instrument range is about 300 meters with an accuracy of 3 mm at 50 meters. The distance between each point and the laser source is calculated according to the time between the emission of the green transmitted pulse and the return signal, that is recorded by a detector.

B. A Real Case

The instrument was used by Enea technicians, INFO-GER unit of Bologna, to scan the archaeological roman site of Juvanum (Montenerodomo, Chieti) [4], in cooperation with the Department of Science of the "G. D'Annunzio" University (Chieti-Pescara).

Fig. 1. ENEA Laser Scanner - Leica Geosystem HDS 3000

The obtained database consists of 16.589.664 points, each designated by x, y, z coordinates, reflectance and color data. Subsequently, the database has been optimized by removing the noise created by trees, metal structures, people in the area and various undesired objects. After this post-processing phase, the final set of points was reduced to 5.665.282 points.

Fig. 2. Juvanum Site (Montenerodomo, Chieti, Italy)

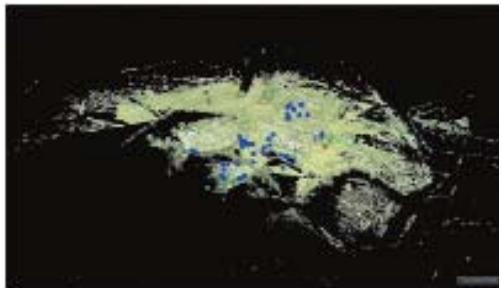


Fig. 3. Juvanum Site (Montenerodomo, Chieti, Italy)

C. Visualization of a set of millions of points

The final database obtained as described above needed to be processed by an application matching the archaeologist needs, i.e.:

- display of a large amount of data in a relatively short time (so allowing the operator to see the cloud of points in a "near real time" manner while he is scanning the archaeological site);
- operations on the points (e.g. distance between any two points, obtain plant views, sections etc.).

Currently there are commercial softwares with the above features, but we were oriented to an "open source" solution using our ENEA-GRID

and the OpenSceneGraph libraries (OSG).

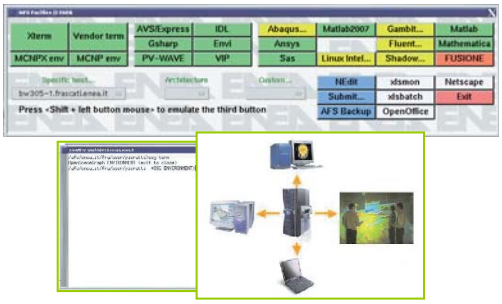


Fig. 4. ENEA-GRID Infrastructure

III. CONFIGURE OPENSCENEGRAPH IN THE ENEA-GRID

A. The OpenSceneGraph Library

The OpenSceneGraph main characteristics are:

- Multiplatform (we have compiled OpenSceneGraph for Linux, Irix, Windows);
- Supports OpenGL and 3D stereo viewer;
- High performance;
- Powerful 3D viewer (osgviewer) that lets us to view in a fairly short time sets of several millions of points.

Any ENEA GRID user has a high level graphic library with the ability to create his own viewer with the desired characteristics. However the OSG does not allow to:

- Measure the distance between two points;
- Extract the plan views;
- Extract the section views.

B. The Point Cloud Viewer

We have faced first the distance issue: the OSG can not allow to pick a single point and memories its coordinates, but this limitation has been overcome by integrating into the viewer code the additional ANN library.
In particular:

- we have built up a database containing the coordinates of all the points from the 3D laser scanner;
- we have superimposed to the displayed image an invisible surface;
- when the user clicks on the desired point the matching coordinates on the surface are kept;
- to face the situation when the user clicks on a display point not associated to any database entry, the point from the original 3D database is assumed to be the point closest to the clicked one;
- the coordinates of the selected point are displayed;

ANN is a library written in the C++ programming language supporting both exact and approximate nearest neighbor searching in space of multiple dimensions. It was implemented by David M. Mount of the University of Maryland and Sunil Arya of the Hong Kong University of Science and Technology [5] [6].

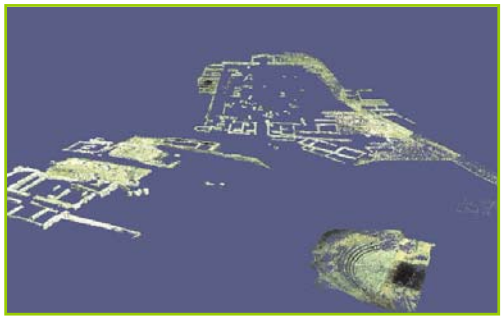


Fig. 5. OSG Point Cloud Viewer

IV. Conclusion

The final 3D point cloud, obtained after the post-processing phase constitutes an objective data base, comprising information about the site morphology, geometric structure and materials of the archaeological structures. The comparison among periodic scanning interventions of the area can give evidence to possible degradation phenomena, and improve conservation and repair activities. Archaeologists can interrogate the database and extrapolate those information directly by means of OSG Point Cloud Viewer using any personal computer without particular characteristic. Specifically, in this case, the obtained data set has been delivered to the Department of Ancient History of the University "G. D'Annunzio" of Chieti-Pescara, as complete suite, application and data-set for operations on point clouds.

Acknowledgments

The authors would like to thank Professor Raffaella Papi and Dr. Patrizia Staffilani of the Department of Ancient History of the University "G. D'Annunzio" of Chieti-Pescara, responsibles of the excavations in the Juvanum archaeological site.

References

[1] Bob Kuehne, Paul Martz, "OpenSceneGraph Reference Manual v2.2," Bob Kuehne, Paul Martz, 2007.

[2] S. Migliori, G. Bracco, P. D'Angelo, "L'architettura di ENEA-GRID," Conferenza GARR 2005.

[3] Leica-Geosystem A G, "Leica HDS3000 3DLaser Scanner,"http://www.leicageosystems.com/hds/en/lgs_5574.htm.

[4] P.Staffilani, "La basilica, Le tabernae, Gli spazi funerari, Il ceto degli equites, in Juvanum. L'area archeologica," Sinapsi Edizioni, Sulmona 2006, 73-80, 109-111, 125-128.

[5] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An Optimal Algorithm for Approximate Nearest Neighbor Searching," Discrete and Computational Geometry, Journal of the ACM, 45 (1998), 891-923.

[6] S. Arya and D. M. Mount,, "Approximate Nearest Neighbor Searching," Proc. 4th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'93), 1993, 271-280.

[7] "Iuvanum. Atti del II Convegno di Studi," (Chieti 31 marzo-1° aprile 1992), Pescara 1996

[8] P. Staffilani, "Iuvanum. Cambiamenti sociali, in New developments in Italian Archaeology," part 2. Papers of Fourth Conference of Italian Archaeology (volume 4), London 1992, pp. 83-91.

Laser Scanner Technologies and 3D Models Integration: Analysis and Methodological Approach

D. Abate[1], M.C. Baracca[2], R. Ciavarella[3], G. Furini[4], S. Migliori[5], S. Pierattini[6]

[1]*ENEA, Bologna, Italy*, dante.abate@enea.it
[2]*ENEA, Roma, Italy*, cristina.baracca@enea.it
[3]*ENEA, Portici(NA), Italy*, roberto.ciavarella@enea.it
[4]*ENEA, Bologna, Italy*, graziano.furini@enea.it
[5]*ENEA, Roma, Italy*, migliori@enea.it
[6]*ENEA, Bologna, Italy*, samuele.pierattini@enea.it

***Abstract*—The 3D scanning laser systems available at ENEA, different for technical characteristics, allow to fulfill a wide range of diagnostic and monitoring requirements in the artistic and cultural heritage field. Since each scanning device is provided with proper file formats and software package for the data management and the reconstruction of the 3D objects acquired, the main problem in constructing complex graphical scenes is to integrate the 3D models preserving all the information detected. Accordingly, the development of Virtual Reality applications requires the analysis of the model file formats and the format conversion tools in order to define a methodology allowing to integrate the different systems and to build 3D scenes navigable by means of VR environments. As example, this paper refers to the development of a virtual museum exhibiting the different 3D models, acquired during various scanning campaigns, navigable by means of VR environments, available through the ENEA Grid.**

***Index Terms*— ENEA, CRESCO Project, GRID Computing, Virtual Reality, Cultural Heritage, Laser Scanner Technology, 3D Models.**

I. INTRODUCTION

The attention of the scientific community towards 3D digitizing techniques applied to cultural heritage is related to the specific needs of conservation: in terms of time and reliability, digital archives of high quality three-dimensional models would constitute a great improvement in respect of the past traditional methods of documentation. Digital archives are durable and unalterable, and thus can be used as reference for degradation monitoring and restoration activities. The scanning laser systems available at ENEA, acquiring 3D objects and detecting information like position, color and reflectance of each point, allow to meet a wide range of diagnostic and monitoring requirements in the artistic and cultural heritage field. Each laser scanning system is provided with a specific software for the data management and analysis, capable to produce 3D models of the scanned objects in specific output file formats. A test case was adopted, in order to define a methodology allowing the inclusion of such 3D models in the same graphical scene. The objective was to develop a "Virtual Museum" in which to set up exhibitions, inserting/removing 3D models produced by different laser scanner system and

to navigate the 3D graphical scene by means of the VegaPrime [1] environment or the OpenSceneGraph [2] toolkit available by means of the ENEA-Grid [3]. The integration of the different laser scanning systems consists in the definition of the conversion procedures of the 3D models, from the specific output file formats to the OpenFlight format (.flt) [4], supported by both the chosen virtual reality environments.

A. Laser Scanner Systems in ENEA

Laser scanning technology allows the digital acquisition of three-dimensional objects as point clouds. The digital geometric description of the object is discrete, the resolution selected for the acquisition defines the density of the point cloud and so the details of the representation. Every point is described by a spatial position in x y z coordinates respect to the origin represented by the position of the scanner. Due to different characteristics of the laser scanner systems in ENEA, it is possible to choose, case by case, the more suitable to be applied and to produce 3D models of objects with size ranging from tens of centimeters to tens of meters.

- **ITR** (Imaging Topological Radar) developed by ArtVisLab-ENEA [5]
  Output file format: PLY (Polygon File Format) with VCG extension library; SW package: ISIS ply viewer;
- **HDS3000** by Leica-Geosystem [6]
  Output file format: DXF, DWG, MDL and PTS; SW package: Cyclone and Cloudworks module for Autocad;
- **VIVID 900** by MINOLTA [7]
  Output file format: STL, DXF, OBJ,ASCII and VRML; SW package: Advantage Polygon Editing Software;
- **Desktop 3D Scanner** by NextEngine [8]
  Output file format: STL, OBJ, VRML and 3DS; SW package: ScanStudio CORE™.

Since each scanning device is provided with proper output file formats and software packages for the data management and the reconstruction of the 3D objects acquired, the main goal in realizing complex graphical scenes is to integrate the 3D models in a VR application, through format conversion preserving all the detected information.

B. 3D Models

The availability of 3D digital models not only does support the promotion and diffusion of the heritage by means of multimedia applications, but also permits to improve many processes like cataloguing, study, preservation, restoration planning and simulation, fragile artifacts monitoring, virtual reconstruction, replica production. 3D laser scanner devices allow the analysis real-world objects or environments, to collect data (point clouds) on its shape and possibly its appearance (i.e. color/reflectance). The collected data can then be used to construct digital, three dimensional models useful for a wide variety of applications, including the exposition in a "virtual museum".

Here below, some 3D models produced by means of:

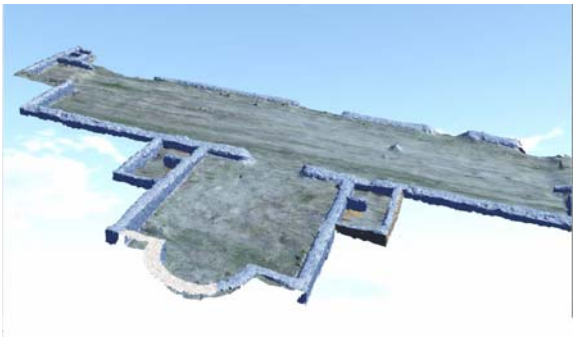
- **VIVID 900**
- **ITR**

Emperor Costantino marble head (Museum, Frascati)

Old soldier and young soldier (Museum Certosa, Bologna)

Polichrome pot 2

Polichrome pot 1



Basilica - Juvanum archaeological site (Montenerodomo, Chieti)

➢ Desktop 3D Scanner

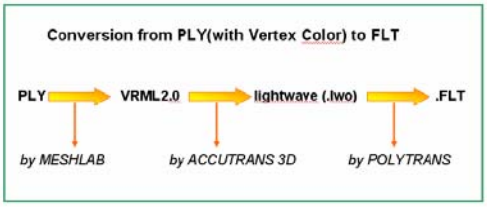
Small artefact

➢ HDS3000

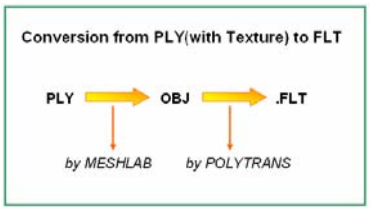
C. Conversion to OpenFlight format

The Open Flight format (.flt) is supported by both VegaPrime and OpenSceneGraph; from the analysis of the 3D models proprietary file formats and the format conversion tools, it was possible to define the 3D models conversion methods to OpenFlight format without loss of information. After analyzing and testing input and output formats of the most common software for 3D model conversion, the following conversion schemes were defined:

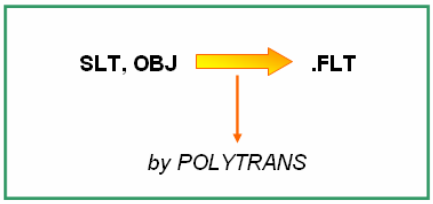
for PLY format:



or



and for STL or OBJ



D. Positioning and scaling models in a 3D scene

Since every scanned object is represented in a 3D space, accordingly to the space configuration of the used scanner system, it is useful to refer all the models to the same space: by means of Accutrans 3D [9] it is possible to redefine the origin for each model and so refer it to the same space. Moreover, due to the different characteristics of the laser scanner systems, two further aspects must be taken in to account:

- Different scaling factor are applied to models originated by different systems;
- The number of points defining the mesh of a model usually is much higher than the one required for a VR application.

LynX Prime, the editing tool of VegaPrime, allows to define the "scale" parameters to be applied along each axis to an .flt model. In this way it is possible to re-establish right proportions to the models.

The number of points in a mesh can be decreased through the application of decimation algorithms. For example, MESHLAB [10] allows to choose among some filter and statistical decimation algorithms.

E. Virtual Museum

In the framework of the ENEA CRESCO project [11], it was decided to develop an application, running on the ENEA Grid, simulating a museum exhibition; that is, starting from the 3D geometrical modeling of a museum architecture equipped with textures, materials, lights, etc., to insert in the graphical scene some of the 3D models produced by means of different laser scanner systems. Once converted in the OpenFlight format, each model was correctly scaled and placed in the navigable museum environment, as shown in the pictures below; the Juvanum archaeological site model can be accessed navigating through a picture of the Juvanum archaeological park [12] placed on an internal wall.

Museum exterior

II. CONCLUSION

Adopted as a test case, the "virtual museum" results in a flexible and dynamic application, allowing to change/insert 3Dmodels produced

Museum interior

by ENEA in different activities involving the use of laser scanner systems. It is a powerful tool for documenting project and collaboration results in the field of Cultural Heritage. Moreover it could be the basis for the development of a 3D multimedia application.

REFERENCES

[1] Presagis Inc., "Vega Prime ," http://www.presagis.com/products/visualization/details/vegaprime/

[2] Bob Kuehne, Paul Martz, "OpenSceneGraph Reference Manual v2.2," Bob Kuehne, Paul Martz, 2007.

[3] S. Migliori, G. Bracco, P. D'Angelo, "L'architettura di ENEA-GRID," Conferenza GARR 2005.

[4] Presagis Inc., "OpenFlight format," http://www.presagis.com/products/standards/details/openflight/

[5] ENEA FIM Dpt, "Laboratorio di Visione Artificiale," http://fim.enea.it/laboratori-e-impianti/laboratorio-di-visione-artificiale

[6] Leica-Geosystem A G, "Leica HDS3000 3DLaser Scanner,"http://www.leicageosystems.com/hds/en/lgs_5574.htm.

[7] Konica Minolta, "3D Measurement Instruments," http://www.konicaminolta.com/sensingusa/products/3d

[8] NextEngine Inc., "Desktop 3D Scanner," http://www.nextengine.com

[9] MicroMouse Productions, "AccuTrans 3D," http://www.micromouse.ca

[10] CNR ISTI Dpt., "MeshLab: an Open-Source Mesh Processing Tool," http://meshlab.sourceforge.net/

[11] ENEA FIM Dpt., "CRESCO: Centro computazionale di RicErca sui Sistemi COmplessi," http://www.cresco.enea.it

[12] P.Staffilani, "La basilica, Le tabernae, Gli spazi funerari, Il ceto degli equites, in Juvanum. L'area archeologica," Sinapsi Edizioni, Sulmona 2006, 73-80, 109-111, 125-128.

Recent Results in Rendering Massive Models on Horizontal Parallax-Only Light Field Displays

M. Agus[1], F. Bettio[1], F. Marton[1], G. Pintore[1], and A. Zorcolo[1]
[1]*CRS4, Pula, Italy, vic@crs4.it*

***Abstract*—In this contribution, we report on specialized out-of-core multiresolution real-time rendering systems able to render massive surface and volume models on a special class of horizontal parallax-only light field displays. The displays are based on a specially arranged array of projectors emitting light beams onto a holographic screen, which then makes the necessary optical transformation to compose these beams into a continuous 3D view. The rendering methods employ state-of-the-art out-of-core multiresolution techniques able to correctly project geometries onto the display and to dynamically adapt model resolution by taking into account the particular spatial accuracy characteristics of the display. The programmability of latest generation graphics architectures is exploited to achieve interactive performance. As a result, multiple freely moving naked-eye viewers can inspect and manipulate virtual 3D objects that appear to them floating at fixed physical locations. The approach provides rapid visual understanding of complex multi-gigabyte surface models and volumetric data sets.**

I. Introduction

MULTISCOPIC visualization is an emerging display technology that aims to reproduce three dimensional scenes by generating observer independent light fields. The accurate reproduction of 3D light fields requires generating a large number of light beams of appropriate origin, direction, and color. Recent advances in 3D display design have demonstrated that interactive high resolution light field display technology is practically achievable. Even though considerations on human vision specifics can drastically reduce the amount of data that has to be encoded in a reconstructed light field, rendering still remains a complex and computationally intensive task, which has limited until very recently the applicability of such displays to presentation of static images, prerecorded movies, or small graphics models.

In this contribution, we report on specialized out-of-core multiresolution real-time rendering systems able to render massive surface and volume models on a special class of horizontal parallax-only light field displays. The rendering methods employ state-of-the-art out-of-core multiresolution techniques able to correctly project geometries onto the display and to dynamically adapt model resolution by taking into account the particular spatial accuracy characteristics of the display. As a result, multiple freely moving naked-eye viewers can inspect and manipulate virtual 3D objects that appear to them floating at fixed physical locations. The approach provides rapid visual understanding of complex multi-gigabyte surface models and volumetric data sets. The rest of the paper is organized as follows: section II briefly describes the design of the light field displays considered, while section III and section IV detail the rendering systems developed to interactively visualize, respectively, massive surface models and scalar volume datasets.

II. Light field display concept

In the light field displays considered here [1], projectors are densely arranged in a horizontal

linear array behind the screen, each one projecting a specific image onto the holographic screen to build up a light field (see figure 1 left). Mirrors located at the sides of the display reflect back onto the screen the beams that would otherwise be lost, creating in this way virtual projectors that increase the display field of view. Each projector emits light beams toward a subset of the points of the holographic screen, so that each screen point is hit by multiple light beams coming from different projectors. The screen, a holographically recorded, randomized surface relief structure, performs selective directional transmission of light beams. Horizontally, the surface is sharply transmissive, in order to maintain a sub-degree separation between views. Vertically, the screen scatters widely, so that the projected image can be viewed from essentially any height. The angular light distribution profile introduced by the holographic screen is characterized by a wide plateau and steep Gaussian slopes precisely overlapping in a narrow region in the horizontal direction. This results in a homogeneous light distribution and continuous 3D view with no visible crosstalk within the field of depth determined by the angular resolution. For building a rendering pipeline on this kind of displays it is necessary to determine where 3D points should be drawn on a given projector to produce a perspective correct image for the viewer. However, the linear perspective is not sufficient, because it ignores the transformation performed by the holographic screen. Since the screen is selective only in the horizontal direction, but scatters widely in the vertical one, the displayed light field's dimensionality is reduced, and the application must decide how to deal with the missing degree of freedom. As explained in [2], the solution consists of using a multiple center of projection (MCOP) approach (see figure 1 right).

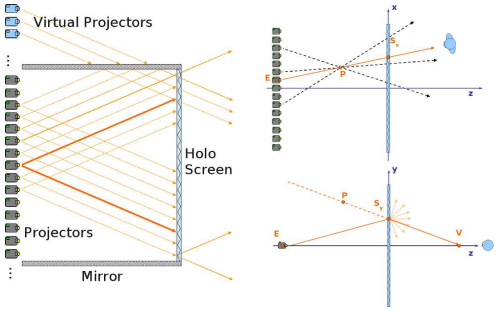


Fig. 1. **Display concept.** Left: Each projector emits light beams toward a subset of the points of the holographic screen. Top right: horizontally, the screen is sharply transmissive and maintains separation between views. Bottom right: vertically, the screen scatters widely, so that the projected image can be viewed from essentially any height, and the MCOP approximation chooses the ray passing through a virtual observer position.

III. Multiresolution Visualization of Massive Models

In this section, we briefly describe our method to drive the light field display in order to get scalable visualization of massive models. The technique is a parallel spatial 3D display-aware version of Adaptive TetraPuzzles [3]. It uses a distributed image generation system implemented on a cluster, with a front-end client PC selecting the level of detail from the multiresolution structure and multicasting graphics commands to back-end PCs. The characteristics of multiresolution techniques based on coarse grained adaptation are exploited to efficiently distribute data to back-end nodes as well as to efficiently pass them to the GPU through preferential paths.

The overall architecture of the rendering system is depicted in figure 2.

The Tetrapuzzles method uses a conformal hierarchy of tetrahedra generated by recursive longest edge bisection to spatially partition the model in a preprocessing step. Each tetrahedral cell contains a precomputed simplified version of the original model. The representation is constructed off-line during a fine-to-coarse parallel out-of-core simplification of the surface contained in diamonds (sets of tetrahedral cells sharing their longest edge). Appropriate boundary constraints are introduced in the simplification process to ensure that all conforming selective subdivisions of the tetrahedron hierarchy lead to correctly matching surface patches

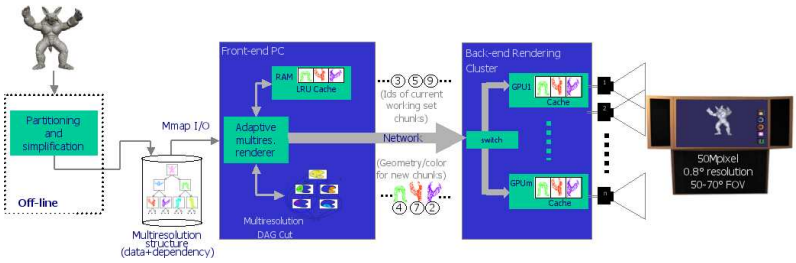


Fig. 2. **Scalable rendering architecture.** A front-end client PC selects the level of detail from the multiresolution structure and multicasts graphics commands to back-end PCs that perform the actual rendering. Object-based communication with extensive caching leads to an efficient implementation.

(see [3] for more details). The main advantage of the method is its ability to rapidly produce seamless variable accuracy reconstructions by assembling precomputed patches.

Since rendering for the spatial 3D display requires the adaptation of vertices to very small (voxel-sized) triangles, controlling triangle shapes during simplification to reduce triangle counts in nearly flat areas is no longer important. Thus, instead of performing high-quality (quadric based) simplification [3], we construct diamonds with a simplification method that produces (roughly) uniformly tessellated meshes, and use edge length as a measure of tessellation accuracy. This approach allows us to manage colored meshes by simply using a color-per-vertex representation. With such a structure, variable resolution rendering is implemented by simple stateless top-down traversals of the binary trees used to encode the tetrahedron hierarchy, which combine view-frustum and contribution culling. The traversal is performed once per spatial 3D frame, and consequently generates as a result the set of patches that needs to be rendered for all the views. Representative video frames are shown in figure 3. The sequences were recorded with a hand held video camera. freely moving in the display's workspace. Note the parallax effects and the good registration between displayed object space and physical space, which demonstrate the multi-user capability of the display. The perceived image is fully continuous. This is qualitatively very different from other contemporary multiview display technologies, which force users into approximately fixed positions, because of the abrupt view-image changes that appear at the crossing of discrete viewing zones [4].

IV. Visualization of Massive Scalar Volumes

In this section, we briefly describe our method to drive the light field display in order to get scalable interactive volume ray casting visualization of huge volumetric datasets. Our technique is derived from a single pass out-of-core raycaster introduced in [5], adapted to work on light field displays similarly to [2]. The overall approach is based on a GPU raycaster, which follows rays generated by a MCOP projection model, while adaptively sampling pre-filtered versions of the dataset at resolutions matching the varying spatial accuracy of the display. In order to allow the volume rendering of very large datasets, we rely on an adaptive technique based on the decomposition of a volumetric data set into small cubical bricks, which are then organized into an octree structure maintained out-of-core. The octree contains the original data at the leaves, and a filtered representation of children at the inner nodes. Each node also stores the range of values, as well as high quality precomputed gradients. In order to efficiently support linear interpolation, we replicate one layer of neighboring samples at each brick boundary. The octree is stored in an out-of-core structure, based on Berkeley

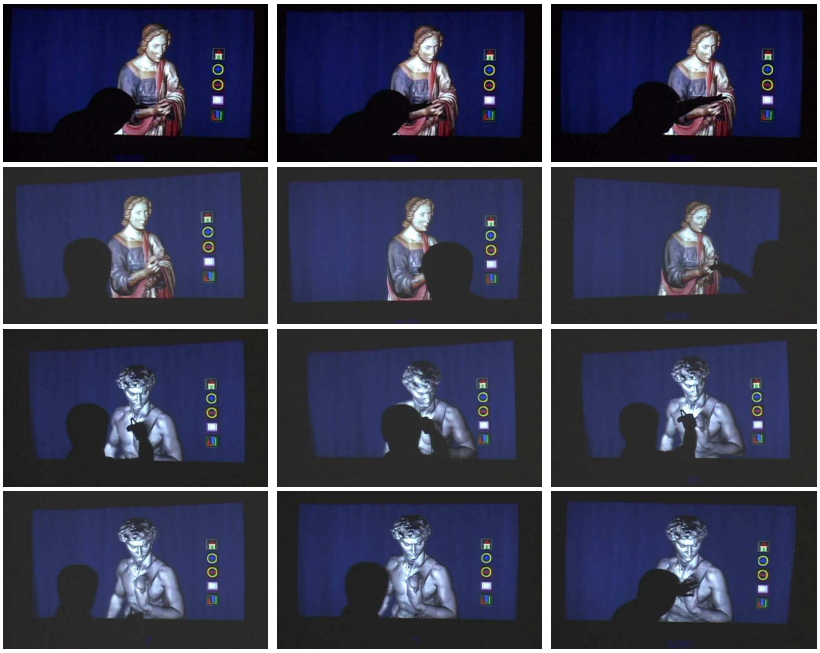


Fig. 3. **Interaction sequence.** These images, taken from the accompanying video, show successive instants of interactive manipulation of the multi-million triangles colored datasets on the 50Mpixel display.

DB, and data is losslessly compressed with the LZO compression library. The system has been developed in order to fulfill all requirements involved in the analysis of high quality and high resolution medical and anatomical data. Volume data are represented as 16 bit scalars, and 32 bit gradients. The gradients are precomputed by employing high quality 5x5x5 Sobel filtering. The system is also able to manage and render segmented datasets represented as labelled voxels. In that case, the precomputation of levels of detail is modified to choose for each value the most popular label instead of the average. At runtime, a working set of bricks is generated and incrementally maintained on CPU and GPU memory by asynchronously fetching data from the out-of-core octree. The working set is created by an adaptive loader on the basis of the current object position in relation to the screen and transfer function. Once the current working set is defined, a compact indexing structure, which spatially organizes the current working set into an octree hierarchy, is encoded in a small texture. This structure spatially organizes the leaves of the current view dependent representation into an octree with neighbor pointers. The inner nodes of this structure simply contain pointers to children, and only the leaves refer to volume data nodes stored in the memory pool. The spatial index structure is exploited by an efficient stackless GPU raycaster, which is able to compute the volume rendering integral by enumerating non-empty bricks in front to back order, adapting sampling density to brick resolution, and stopping as soon as the accumulated opacity exceeds a certain threshold, updating both the framebuffer and the depthbuffer. The prototype volume ray caster implements a number of composition strategies, including Direct Volume Rendering with a Phong illumination model, boundary enhancement and view dependent transparency [6]. When dealing with labeled data, trilinear filtering is substituted with nearest neighbor to preserve label values [7]. In order to drive the light field display the scene is rendered once per projector view. The

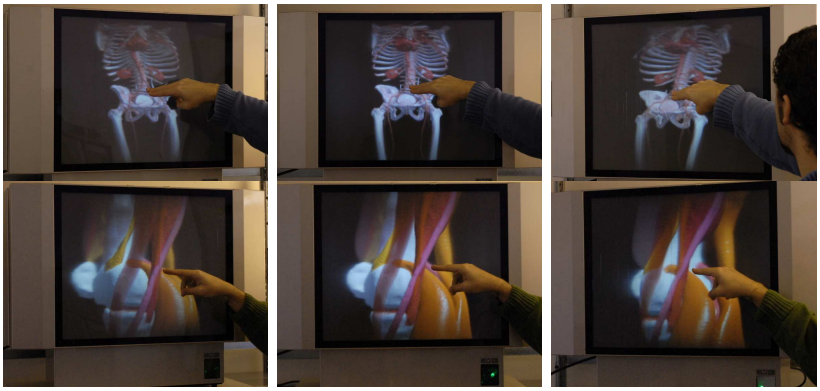


Fig. 4. **Inspecting raw and segmented datasets on the light field display.** Top: Raw whole body contrast CTA acquired on a 16 detector CT scanner (source: Radiology Department, University of Geneva). The volume has a resolution of $512 \times 512 \times 1559$ with 16 bit/sample. Bottom: segmented leg reconstructed from MRI acquisitions (source: MiraLab, Geneva). The volume has a resolution of $404x474x2050$. Pictures are taken with a hand-held camera at different viewing angles, in order to highlight the horizontal parallax of the light field display. Images are rendered at about 6.5fps.

resulting system is capable to interactively explore extremely large datasets on light field displays (see figure 4 for some examples). The prototype can be considered a testbed in a development process aimed at creating a really good 3D light field based radiology workstation in the future [8]. Perceptual evaluation tests have been carried out and tend to prove that the system provides correct depth cues, helps in layout discrimination and is clearly superior to two-dimensional displays for path tracing tasks, common in operations such as the understanding of vascular structures. The first feedback received from physicians and radiologists seems to confirm this fact. Future work aims at performing an evaluation of the system for diagnostic tasks in a clinical context.

REFERENCES

[1] T. Balogh, T. Forgcs, T. Agocs, O. Balet, E. Bouvier, F. Bettio, E. Gobbetti, and G. Zanetti, "A scalable hardware and software system for the holographic display of interactive graphics applications," in *EUROGRAPHICS 2005 Short Papers Proceedings*, 2005.

[2] M. Agus, E. Gobbetti, J. Iglesias Guitián, F. Marton, and G. Pintore, "Gpu accelerated direct volume rendering on an interactive light field display," *Computer Graphics Forum*, vol. 27, no. 3, pp. 231–240, 2008, proc. Eurographics 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2008:GAD'

[3] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, "Adaptive tetrapuzzles: Efficient out-of-core construction and visualization of gigantic multiresolution polygonal models," *ACM Trans. on Graphics (SIGGRAPH 2004)*, vol. 23, no. 3, pp. 796–803, 2004.

[4] W. Matusik and H. Pfister, "3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 814–824, Aug. 2004.

[5] E. Gobbetti, F. Marton, and J. Iglesias Guitián, "A single-pass GPU ray casting framework for interactive out-of-core rendering of massive volumetric datasets," *The Visual Computer*, vol. 24, no. 7-9, pp. 797–806, 2008, proc. CGI 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Gobbetti:2008:SGR'

[6] S. Bruckner and M. E. Gröller, "Style transfer functions for illustrative volume rendering," *Computer Graphics Forum*, vol. 26, no. 3, pp. 715–724, Sept. 2007.

[7] M. Agus, A. Giachetti, E. Gobbetti, J. Iglesias Guitián, and F. Marton, "Towards advanced volumetric display of the human musculoskeletal system," in *Eurographics Italian Chapter Conference*. Conference held in Salerno, Italy: Eurographics Association, June 2008. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2008:AVD'

[8] M. Agus, F. Bettio, A. Giachetti, E. Gobbetti, J. A. I. Guitián, F. Marton, J. Nilsson, and G. Pintore, "An interactive 3d medical visualization system based on a light field display," *The Visual Computer*, vol. 25, 2009, to appear. [Online]. Available: http://www.crs4.it/vic/cgi-bin/bib-page.cgi?id='Agus:2009:'

VTA: Grid-Based Thermal Images Processing for Volcanic Activity Monitoring

S. Coco[1], L. Drago[1], A. Laudani[1], L. Lodato[2],
S. Mangiagli[2], G. Pollicino[1] and O. Torrisi[2]
[1]*DIEES, University of Catania, Catania, Italy*
[2]*INGV Sezione di Catania, Catania, Italy, mangiagli@ct.ingv.it*

***Abstract*—In this paper the authors present a new program, developed in IDL, denominated Volcano Thermo Analysis, tailored to the thermal images processing and to the volcanic activity monitoring. VTA was appositely developed in order to extract from all the temporal series of images available a set of important features, able to characterize with a good approximation the volcanic activity and evolution. Since the analysis of this data requires a lot of CPU and storage resources, the porting of VTA on GRID environment has been successfully performed. This has allowed to perform faster analysis on huge set of data, proving VTA GRID as a really useful instrument for scientific research.**

***Index Terms*—Grid Computing, Volcanic Activity Monitoring, Thermal Images.**

I. Introduction

THE study of temperature changes and gas composition at fumaroles can provide insights into the thermodynamics of a volcano's magmatic and hydrothermal system, as well as the way in which the two systems interact [1], [2], [3]. Consequently, gas flux, composition and temperature are often routinely or continuously measured and mapped at active fumarole fields. In the last years this monitoring also makes use of non-contact infrared thermometer and thermal cameras [4], [5]. In particular, thermal images are a very useful mean, since they allow the researcher to perform both qualitative and quantitative analysis; in fact from a qualitative point view it is possible to map lava flow and to individuate any possible fractures within the volcanic chamber, whereas from a quantitative point view it is possible to estimate data about the flux and the changes in the temperature of lava, the effusion rate, etc.. The increasing use of thermal cameras is also due to some other important advantages with respect the traditional adopted monitoring techniques. In fact, temperatures traditionally have been measured using thermocouples installed at a few fixed points. This fact implies that the instruments could work in harsh conditions because of temperature and the presence of corrosive agents. In addition, permanent installations only allow a few fumaroles to be targeted, when many are active over a widely dispersed area. The highest temperature fumaroles are usually targeted, but redeployment is necessary if new fumarole zones develop or if the existing targeted zones decay. Non-contact infrared thermometers and thermal cameras allow many or all vents over a fumarole field to be measured remotely. Furtherly thermal cameras offer an advantage over infrared thermometers by providing a synoptic perspective, i.e. an image (or series of images) can cover the entire field to include all fumaroles within a single image or image mosaic. This can be achieved by targeting the fumaroles from a good vantage point outside of the field (see for example 1).

Since 2001 the INGV-CT Section has used

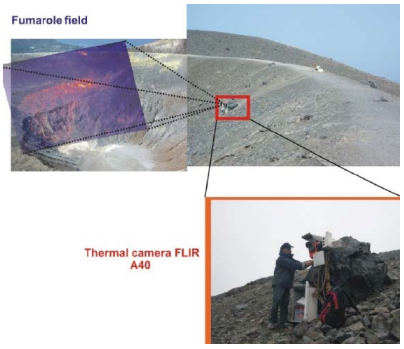


Fig. 1. Fumarole Field and FLIR thermal camera installation between 25 and 27 June 2006

Fig. 2. Visible picture of studied fumarole field

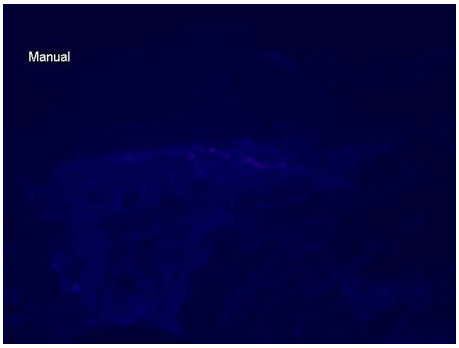


Fig. 3. Image from thermal camera before pre-processing

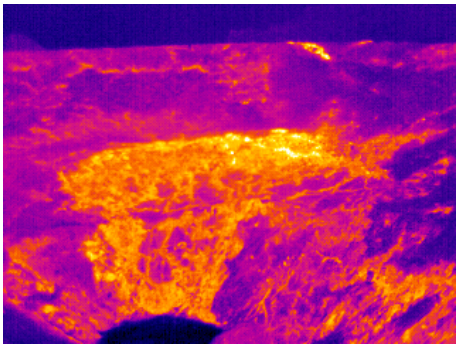


Fig. 4. Image from thermal camera after pre-processing

thermal images for the monitoring of volcanoes Etna, Stromboli, Vulcano e Panarea [7]. In this paper the authors present the development and the porting in the Grid Environment of a software appositely implemented for thermal images processing. This dedicated software developed in IDL and denominated Volcano Thermo Analysis (VTA) can extract a set of important features, able to characterize with a good approximation the volcanic activity. Since the processing of long temporal series of images is a very time consuming task, the porting of this application in the Grid environment makes it possible to perform faster analysis of huge data set.

The paper is structured as follows: in section II the installation of the thermal camera designed to monitor volcanic activity of "Vulcano" island is presented; in section III the main characteristics of the VTA software are shortly described; in section IV the porting of the VTA software in the Grid environment and the results obtained are reported; authors' conclusions follow in section V.

II. Volcanic Activity Monitoring by Thermal Images Processing: Thermal Camera Installation and Data Acquisition

On June 2006 a Forward Looking Infrared (FLIR, Model A40) thermal camera was installed in Vulcano's La Fossa Crater (fig. 1). the camera was positioned in the opposite side of the inside of the crater, 400 m away from the fumarole. The camera takes a not cooled bolometer and operates in the $7.5 - 13\ \mu m$ waveband, producing a calibrated 240×320 pixel image. In figg. 2-4 the same fumarole field is shown

respectively in a visible picture, in a not pre-processed thermal image and in a pre-processed thermal image. The camera was mounted on a tripod. Both in-situ humidity and temperature measurement [6] are used for LOWTRAN-standard atmosphere-based atmospheric correction of all pixel temperatures using the camera's inbuilt correction algorithm. The distance of the line of sight from this vantage point to the lower zone, middle zone, and rim is 350, 420 and 470 m, respectively. These line of sight distances give pixel sizes of 0.46 m at the crater floor, 0.55 m at fumarole and 0.61 m at the rim.

A. *Methodology of data acquisition*

Thermal images and meteo data are acquired in our site in Lipari and are transferred over a GARR-TCP/IP connection and are archived in Catania CED. Ambient temperature, relative humidity, DCF time, rainfall rate are acquired by the meteo station and directly put within the thermal camera software for applying the atmospheric corrections each 30 min. The thermal camera started to work in September 2006. The frequency of acquisition is one frame every 30 minutes. The first two-year of data on temperature distribution frequency were recorded with this new methodology of acquisition, and automatically elaborated by software at INGV-CT Section.

III. The Software Volcano Thermo Analysis (VTA)

A dedicated software written in IDL (Interactive Data Language), denominated Volcano Thermo Analysis (VTA), was appositely developed in order to extract a set of important features, able to characterize with a good approximation the volcanic activity. The aim of this software is to develop a tool for thermal classification of fumaroles before extracting best-condition pixel-integrated temperatures for each classified zone. In fact the thermal image allows to define regions of fumaroles, in order to perform an examination of the thermal characteristics of each region of interest (ROI).

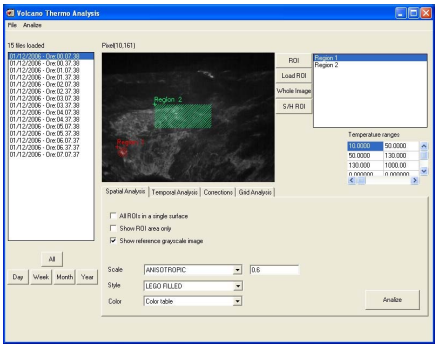


Fig. 5. VTA Graphical User Interface

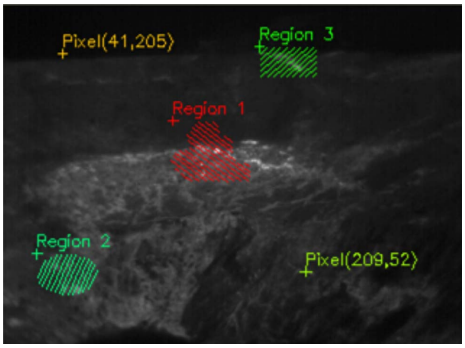


Fig. 6. Different Region Of Interest (ROIs) definition

In addition the VTA is able to analysis all the temporal series of images available in order to achieve the time-event analysis and the dynamic of the volcanic evolution. All the VTA functions are interactively managed and executed by GUI (fig. 5), which controls all the processing operations. In particular the program first loads and opportunely converts the thermal images, then, according to the Region Of Interest (ROI) and the temperature ranges defined by the user, provides automatically spatial and statistic analysis.

Three main steps should be executed before running the Volcano Thermo Analysis (VTA) software:

- Data loading: the thermal images can be loaded by selecting the images belonging to the same period considered (Day, Week, Month, Year) or by defining any desired time and date.
- Regions Of Interest (ROI) definition (see

fig. 6): it is possible to define an unlimited number of ROIs whose shape can be regular (rectangular, elliptical, circular, punctiform, etc.) or irregular (defined by using the mouse pointer); the defined ROIs can be saved and reloaded for further processing.
- Temperature ranges definition: up to 10 different temperature ranges can be defined in order to filter data (only points inside these ranges will be considered).

The output of the VTA consists of texts and images which report the information desired, in particular:

- Spatial localization info on single image or on multiple images (figg. 7-9):
  1) ROI indication.
  2) Ranges indication.
  3) Max and min values.
  4) Vertices height proportional to pixels temperature.
  5) Colorbar indicating temperature.
  6) Reference greyscale image.
- Temporal trend (fig. 10): maximum and average temperature plot related to each selected ROI or to the whole image, with one point for each image, hour, day, week, month, year
- Statistical info (spatial or temporal)
  1) Max, average and min values for each ROI and for the whole image.
  2) Frequency histograms (fig. 11) are generated for each image and for temporal series (in a specifed time interval or related to hour, day, week, month and year)

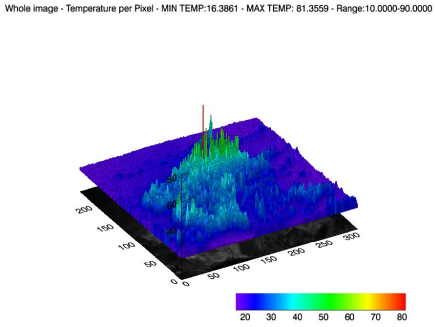


Fig. 7. VTA output: result of analysis of a whole image for a large temperature range

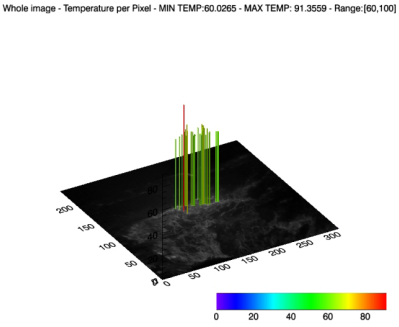


Fig. 8. VTA output: result of analysis of a whole image for a narrow temperature range with reference greyscale image

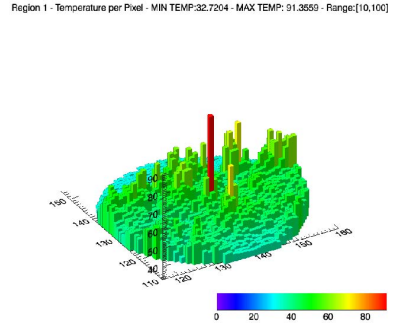


Fig. 9. VTA output: results of analysis of an elliptical ROI

IV. Porting VTA to the GRID environment

The analysis of this data requires a lot of CPU and storage resources and this represents a serious limitation, and often this can overwhelm the performance capability of a workstation. For example, considering an image every 30 minutes we have an amount of data of about 14 MB for a single day, about 100 MB for a week, 400 MB for a month and 5 GB for a year. Pratically it was not possible to run on a workstation equipped with a Intel Core2Duo 2.13 GHz CPU and 2 GB RAM an analysis with more than one month data. Fortunately

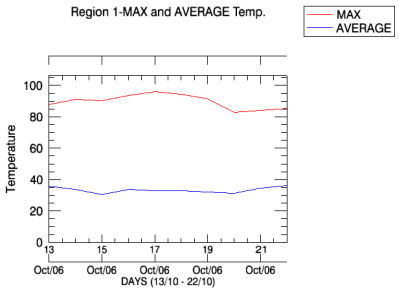


Fig. 10. VTA output: temporal trend of maximum and average temperature of the whole fumarole field in the period 13/10/06 - 22/10/06

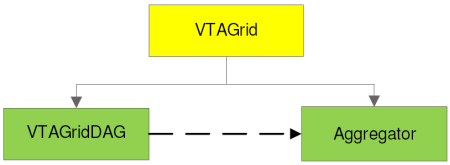


Fig. 12. VTA GRID as a DAG

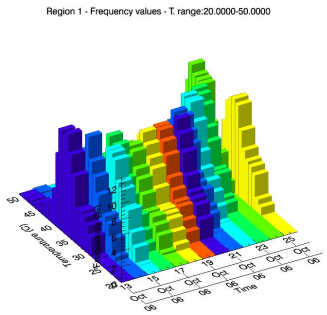


Fig. 11. VTA output: frequency histograms

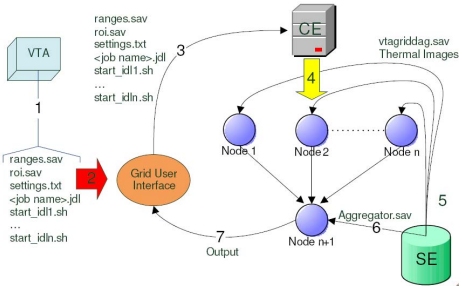


Fig. 13. Flow diagram of VTA GRID

the INGV and the University of Catania are involved in a project for the development of a GRID infrastructure (a virtual supercomputer created by using a network of independent, geographically dispersed, computing clusters which act like a grid [8]) and of software for this GRID. Consequently we have studied the possibility to port VTA to the GRID. Being VTA developed in IDL, the first problem was to use IDL in Grid environment. In fact the VTA was developed by considering an interactive approach, whereas in the Grid infrastructure the IDL must be used by command line. The first step was to separate the preprocessing (data input) module and postprocessing (data output and results visualization) module from the thermal images processing module, in such a way to adapt VTA to run in batch mode. The second step was to upload to the Grid the thermal images, since VTA works with a huge number of images. In addition in order to better use the Grid infrastructure the VTA Grid version has been conceived considering the application as a Direct Acyclic Graph (DAG): the analysis task is first subdivided in the maximum number of machines available and then another part of the program provided the aggregation of the results (see figg. 12,13).

TABLE I
VTA GRID VS VTA PC

1 month of data on a PC	20 minutes
2 month of data on Grid	15 minutes
4 month of data on Grid	40 minutes

A. VTA on GRID results

The result in term of time execution obtained in the Grid environment are summarized in Table I. In particular, the comparison regards the processing of 1 month of data on a PC Intel Core2Duo 2.13 GHz, with 2 GB RAM, and the processing of 2 and 4 months of data on the Grid environment, by using a DAG with 2 intermediate nodes. It is worth noticing that it was not possible to run 2 month of data on the PC. Unfortunately the jobs execution time is not perfectly predictable on the Grid environment since the total jobs execution time is equal to Jobs handling time and effective running time, so we refer in the Table I to average value. The gain in the performance of VTA by using Grid computing results however considerable.

V. Conclusions

A new method for Volcanic Activity Monitoring by mean of Thermal Images has been adopted at INGV-CT section, thanks to the development and set up of a specific software (VTA) developed in IDL. The porting of this software in the GRID environment has been experimented with success. GRID computing has greatly enhanced VTAs potentialities, allowing us to perform faster and multiple analysis on huge set of data, proving itself as a really usefull instrument for scientific research.

Acknowledgment

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.).

References

[1] Aiuppa A, Federico C, Giudice G, Gurrieri S (2005) Chemical mapping of a fumarolic field: La Fossa Crater, Vulcano Island (Aeolian Islands, Italy). Geophys Res Lett 32:L13309 DOI:10.1029/2005GL023207

[2] Harris AJL, Stevenson DS (1997a) Thermal observations of degassing open conduits and fumaroles at Stromboli and Vulcano using remotely sensed data. J Volcanol Geothem Res 76:175-198

[3] Harris AJL, Stevenson DS (1997b) Magma budgets and steady-state activity of Vulcano and Stromboli volcanoes. Geophys Res Lett 24:1043-1046

[4] Harris AJL, Maciejewski AJH (2000) Thermal surveys of the Vulcano Fossa fumarole field 1994-1999: evidence for fumarole migration and sealing. J Volcanol Geotherm. Res 102:119-147

[5] Chiodini G, Vilardo G, Augusti V, Granieri D, Caliro S, Minopoli C, Terranova C (2007) Thermal monitoring of hydrothermal activity by permanent infrared automatic stations: Results obtained at Solfatara di Pozzuoli, Campi Flegrei (Italy). J Geophys Res 112: B12206 DOI:10.1029/2007JB005140

[6] Ball, M., and H. Pinkerton (2006), Factors affecting the accuracy of thermal imaging cameras in volcanology, J. Geophys. Res., 111, B11203, doi:10.1029/2005JB003829.

[7] Harris AJL, Lodato L., Dehn J, Spampinato L: Thermal Characterization of the Vulcano Fumarole Field, submitted to Bulletin of Volcanology (2007)

[8] I. Foster, C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, 1999.

Implementation of a Limited Area Ensemble Meteorological System in a Grid Environment

Marino Marroccu[1], Gabriella Pusceddu [1], Elisaveta Peneva[1] and Antioco Vargiu[1]
[1]*CRS4, Parco Scientifico e Tecnologico, Edificio 1, 09010, Pula, Italy*
{marino,gabri,elfa,antyv}@crs4.it

***Abstract* — Climate changes due to global warming in the Mediterranean basin, ever more frequently, generate extreme meteorological events such as violent extra tropical storms characterized by wide-spread flooding and/or gale force winds. For this reason, there is an increasing demand for reliable probabilistic short range weather forecast. At regional scale these kinds of forecasts are even more valuable especially in cases of high impact weather that may bring about loss of lives and properties. To estimate the probability of a certain event to develop the Ensemble Prediction (EP) approach, in which single deterministic forecasts are combined using specific statistical techniques, is frequently used. These EP simulations are computationally very demanding, requiring a great amount of CPU time and large storage capacity. Moreover the run of an ensemble of forecasts is a time critical task which has to be completed in less than a couple of hours to be usable in an operational framework. In this short paper we discuss the results obtained in one of the WPs of the CYBERSAR project, financed by the Italian Ministry of Research (MIUR). A limited area ensemble system, based on the analysis and forecast of the 51 members of the European Centre for Medium Range Weather Forecast (ECMWF) EP System (EPS), has been ported on a GRID of the CYBERSAR project, using the gLite middleware, and tested for a couple of extreme events that hit the Sardinia region in the last months of 2008.**

***Index Terms*— meteorological forecast, EPS, LAM, Grid, gLite.**

I.INTRODUCTION

DUE, to its central position in the Mediterranean Sea and to the effects of Global Warming, the Island of Sardinia, in recent years is ever more frequently experiencing intense rainfall events which have triggered serious damages with strong economical and societal impact. For example in Sardinia floods are, among natural disasters, the more important cause of loss of lives. The aim of this project was the study and the development of computational techniques to simulate the evolution of the probability distribution function of the atmospheric states, since this type of forecast is particularly useful for predicting extreme weather events. In the CYBERSAR project we intended to use the potential offered by the Grid computing technology [3], to run, simultaneously, an ensemble of equally likely forecasts in order to be able to characterize the uncertainty of the single deterministic forecast. The aim was to make more accurate predictions, from a time of forecast of 6 hr till 72 hours, at the spatial scale of about 18 km, and with the constraint of a total time available to complete all the integrations of the order of 2 hours. Two different methodologies, and the related technological aspects, have been tested: the Multi-Analysis Multi-Model (MAMM) approach [3] and the downscaling of the 51

members of the ECMW EPS. Here for reasons of space only this last experiment will be discussed. In detail, in section A we describe the EPS used, in section B we illustrate the implementation of this EPS on the Grid of the CYBERSAR project, in section C one of the meteorological events studied is discussed and finally in section II some results are discussed and conclusions are given.

A. EPS Description

Atmospheric dynamics is described by a set of non-linear partial differential equations: momentum equations, mass conservation, energy conservation, air state equation and water balance. A Numerical Weather Prediction (NWP) model is a specific solver for those equations, with significant simplifications, combined with complex parameterizations of the smallest scale phenomena (sub-grid processes). NWP models are classified into Global Models (GCM) and Limited Area Models (LAM).

The former perform integrations over the entire globe, with a typically spatial resolution of tens of kilometers.

The latter models allow to study limited regions of the globe with higher spatial resolution and more detailed physics parameterizations.

They are, necessarily, driven by initial and boundary conditions from a run of a GCM. The correct representation of initial condition (analysis) is crucial for the integration of a model. Since meteorological analysis are obtained from previous model simulations, corrected and optimized by Data Assimilation (DA) techniques [4], they contain, model and measure, errors which tend to increase in time and lead to divergence of the model from real atmospheric state. To extract useful information for a system with chaotic behaviour, like the atmosphere [5], the probabilistic techniques could be efficiently applied to the numerical weather prediction. In general, these methods analyze the results from different scenarios, obtained using either different models, or different meteorological analysis as initial conditions. In this study we present a chain of procedures to perform a probabilistic forecast on a limited area of the globe, using the ECMWF EPS. This EPS is made of 51 different independent global forecasts. These forecasts are obtained perturbing the initial condition of the "optimal" analysis (control run), in such a way to be consistent with the uncertainty of the initial data and to represent the atmospheric instabilities which have most significant impact on the short-to-medium weather forecast. The atmospheric state predicted with these different ensemble members evolves on trajectories, either convergent or divergent and it is, thus, possible to estimate the probability distribution of meteorological variables in general, and the probability of extreme events to occur in particular. The EPS system we have implemented is shortly described in the next section (see also Figure 1) and the limited area model domain can be seen in Figure 2. Horizontal resolution of our integrations was about 18 km.

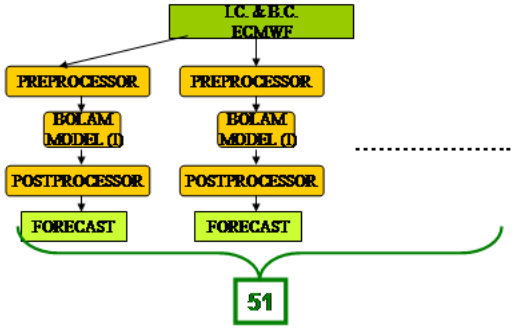


Figure 1. Schematic representation of the ensemble forecast workflow. For a short description see the text in section B.

B. EPS on gLite

In the framework of the CyberSAR project the EPS described in section A has been ported on a gLite platform. The software builds the meteorological ensemble following several steps: EXTRACTION of the data, for the

limited are domain, from the ECMWF archive; PREPROCESSOR: horizontal and vertical interpolation of this data into the model grid and insertion of the static information (orography and vegetation); generation of the 51 different initial conditions and boundary condition for RUN of the BOLAM model; POSTPROCESSOR, to eliminate the redundant information and obtain the data necessary for the ensemble prediction FORECAST (see Figure 1).

The software is installed in a shared grid zone, replicated in each Computing Element (CE). It has also been necessary to install in each CE two additional libraries, named GRIB_API and Jasper, both distributed by ECMWF, needed for decoding data in GRIB2 format.

The critical points of this system are essentially two: the large number of simultaneous model integrations and the large storage capacity required by both input and output data. In order to facilitate the second point we have divided the input data in two groups – analysis and static data. Analysis data (about 800 Mb) are stored in a web server, and the archive of static data which consists mainly of the Earth orography (about 3Gb) is copied in the same shared grid zone as the executable files. On every computing element at the beginning of the simulation the analysis data are downloaded from the web server into the local working zone. The static data, instead, are accessed every time the program needs them.

The execution of the software on the GRID environment is managed by a shell script which needs as an initial parameter a number to define the ensemble member. This parameter has an initial value of 0 for the control run and 1 to 50 to identify each of the perturbed members. A configuration like this is run under gLite using the parametric job option that allows to send n different independent jobs on n computing elements with a unique job descriptor (a single jdl file).

Thus, the data flow, for initialization, for each job is about 16 Mb and the inputbox consists only of the executable parametric script. After this the 51 different jobs are simultaneously executed, and each one generates the corresponding output data in about 50'. The total final output takes about 25GB of disk space and it is managed by the catalog. Each single job generates about 500 Mb of output which are registered in the catalog and moved on the local storage element, increasing the network traffic. The Cybersar gLite GRID, were the jobs were run, consists of 4 nodes geographically distributed.

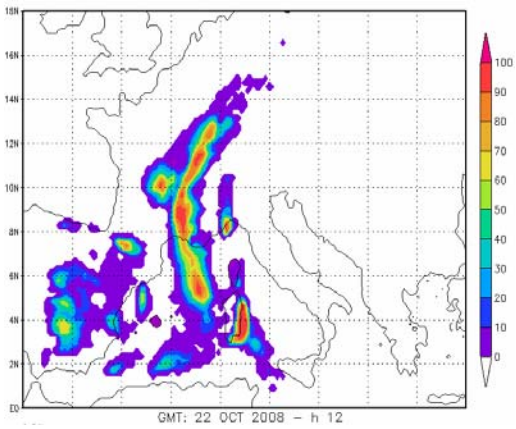


Figure 2. Probability of 3hours cumulated precipitation exceeding 8 mm/3hr, as obtained by the T_0+48hr LAM-EPS forecast. Precipitation is cumulated between 9 UTC and 12 UTC of the 22th October 2008. The colors represent the probability thresholds.

C. Case study

From a meteorological point of view the event we will discuss was due to low-level advection of warm and moist air from southeast and to the simultaneous upper-level advection of cold and wet air into the cyclonic system that invested the Mediterranean basin. Over Sardinia, this strong vertical gradient triggered deep convection.. The strong orographic forcing induced by the mountains of south-eastern Sardinia, coupled to the heat fluxes due to the relatively warm Mediterranean sea, further strengthened the convective system and intensified precipitation. As a result, intense

rainfall hit the southern and eastern coast of Sardinia on the morning of October the 22th 2009 causing severe disruption, heavy damages and the loss of 5 lives. Observed rainfall had had two main temporal peaks: from 5 am to 10 am local time in the southern part of the Island, and from the 8 am to the 12 am in the eastern coast. In the south, near the Capoterra village, where the main damages occurred, at the Poggio dei Pini raingauge station, an hourly rain rate of 175 mm/h and a cumulative value of 372 mm in 5 hours, have been recorded.

II. Conclusions

From the meteorological point of view the results were encouraging. Figure 2 shows the map of probability for 3hr cumulated precipitation (between 9 UTC and 12 UTC of the 22th October 2008) exceeding 8 mm/3hr, as obtained by the T_0+48hr LAM-EPS forecast. As it can be seen the maximum of probability is located in the south-eastern part of Sardinia exactly where the event was measured (see end of the Section C).

From the technological point of view results were good as well since the execution of all the forecast required about 1 hour of computing time (10' for pre-processing and 50' for running). As already mentioned, the critical point for this application was the download of the large amount of output data. We avoided this problem moving only the fields (in our case 3hr cumulated precipitation) needed to calculate the precipitation probability of the ensemble.

Acknowledgments

The authors acknowledge support from the Italian Research Ministry (Projects CYBERSAR MIUR FESR, FDR D.D. 253/RIC. 14/02/2006 and GRIDA3 n. 1433/2006 MIUR FAR D.L. 297/99) and from the Sardinian Regional Authorities.

References

[1] Buzzi A, Fantini M, Malguzzi P. and Nerozzi F., "Validation of a limited area model in cases of Mediterranean cyclogenesis: surface fields and precipitation scores." Meteorology and Atmospheric Physics, 1994, 53: 137-153.

[2] Chessa, P., G. Ficca, M. Marrocu and R. Buizza, "Limited-Area Probabilistic Prediction System for Heavy Rainfall Events." Weather and Forecasting, 2004, 19: 566-581.

[3] Foster I. and Kesselman C., "The Grid 2: Blueprint for a New Computing Infrastructure." 2004, Morgan Kaufmann Publishers, Second Edition.

[4] Kalnay E., "Atmospheric modeling, data assimilation and predictability." 2003, Cambridge university press.

[5] Lorenz E., "Deterministic Nonperiodic Flow" Journal of the Atmospheric Sciences, 1963, 20:130-141.

[6] Marrocu,M. and P. Chessa, "A multi-model/multi-analysis limited area ensemble: calibration issues." Meteorol. Appl. 2008, 15: 171-179.

Marino Marrocu. Currently Head of the Environment Science Group of CRS4. The main area of interest is the development and application of numerical models and other computational tools to analyze environmental data and to simulate the dynamics of hydrological and meteorological systems. The principal personal research topic is the study and application of the Ensemble Prediction technique to mesoscale dynamics.

Gabriella Pusceddu. Graduated in Physics at the University of Cagliari. Since 2000 she works as researcher at CRS4. Currently working in the Environment Science Group since January 2006. Her interest is related to the application of limited area models for the prediction of extreme events and the implementation of ensemble prediction systems. She has developed in recent years competences of GRID computing using the gLite middleware.

Implementation of a Two-Way Grid Refinement Technique for the MITgcm: an Application to the Mediterranean Sea

G. Sannino[1], V. Ruggiero, [2], A. Carillo[1] and V. Artale[1]
[1] *ENEA, via Anguillarese 301, 00123 Roma, Italy, gianmaria.sannino@enea.it*
[2]*CASPUR, via dei Tizii 6, 00185 Roma, Italy, vittorio.ruggiero@caspur.it*

***Abstract*—A two-way Grid refinement technique for the MITgcm model is described. To verify the method the grid-refinement is applied to the real case of the Strait of Gibraltar. The method consists in embedding a finer resolution model of the Strait of Gibraltar in a coarser model of the entire Mediterranean Sea. The two models are coupled via an external driver that performs also all the necessary interpolations. To strongly reduce the communication cost between the driver and the two parallel models, a parallel version of the driver has been developed. The entire nested system is integrated for a multi-decadel period reproducing the actual climate. An analysis of the results obtained has revealed that the nested model is able to reproduce most of the main characteristics of the Mediterranean circulation confirming the robustness of the adopted grid-refinement procedure.**

***Index Terms*—Strait of Gibraltar, Two-way Grid Refinement, Mediterranean model.**

I. Introduction

A Key component of the Mediterranean ThermoHaline Circulation (MTHC) is represented by the Strait of Gibraltar, that with its complex system of contractions and sills (Figure 1) contributes to determine the final magnitude and hydrological properties of the MTHC ([1], [2]). In the last fifteen years several numerical models have been implemented to reproduce and understand the Mediterranean ThermoHaline Circulation (MTHC) and its variability. The main features of the MTHC were initially reproduced by coarse resolution models, characterized by a horizontal grid space of 20-25 Km (among others see for example [3], [4], [5], [6]). Further understanding regarding the physical mechanisms driving the mesoscale variability of the Mediterranean circulation has been achieved only recently by means of eddy-permitting/resolving models having a horizontal grid space of 5-15 Km (among others see for example [7], [8], [9], [10]).

However, even considering only these last modeling efforts, the horizontal resolution cannot be yet considered sufficient to fully reproduce both hydrological and hydraulic properties of the two-way circulation in the Strait of Gibraltar. In fact, in most of the previous models the Strait is approximated by a rectangular pipe having a cross-section wider than the real dimension, in general more than double, and a

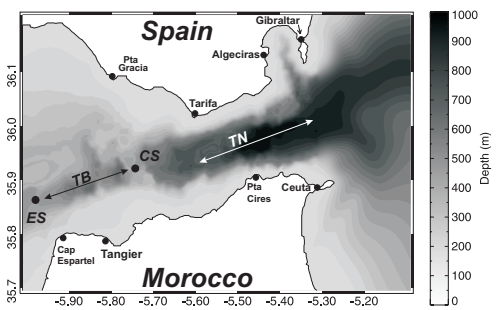


Fig. 1. Map of the Strait of Gibraltar showing the main topographic features: ES is Espartel Sill, TB is Tangier Basin, CS is Camarinal Sill, TN is Tarifa Narrows.

flat bathymetry.

To solve this problem we propose of applying a two-way grid refinement in the region of the Strait of Gibraltar in an eddy-resolving Mediterranean model. This technique consists in embedding a fine-resolution model inside a coarser-resolution model and use the fine-resolution model variables to replace the coarse-resolution model variables over the subregion where the two grids overlap.

The implementation of this technique can be obtained in different ways. In the present paper we develop an implementation that is characterized to be relatively simple and efficient in terms of computational time.

The paper is organized as follows: the model is presented in section II, in section III is described the grid-refinement technique, in section IV are shown the model results, while section V concludes the paper.

II. Nested model

The nested model consists of two models: a coarse grid model (hereinafter CGM), covering the entire Mediterranean Sea and the Gulf of Cadiz (Figure 2), and a fine grid model (hereinafter FGM), embedded within the CGM, whose domain is limited to the Strait of Gibraltar and part of the Alboran Sea at east ($2°33.75'$ W), and part of the Gulf of Cadiz at west ($8°21.25'$ W), see Figure 3 for details). Both models are based on the MITgcm developed by [11], [12]; in particular in the present work the MITgcm is used in its hydrostatic, implicit free-surface, partial step topography formulation ([13]).

The CGM component of the nested model has a resolution of $\frac{1}{8}^\circ \times \frac{1}{8}^\circ$, and so can be considered as an eddy-permitting model. Horizontal viscous and diffusive terms are modeled with a bi-harmonic formulation with diffusivity and viscosity coefficients equal to 1.5×10^{10} m^4 s^{-1}. Vertical eddy-diffusivity is modeled via a Laplacian formulation with diffusivity coefficient ranging from 3.0×10^{-5} m^2 s^{-1} at surface to 1.0×10^{-7} m^2 s^{-1} at the bottom, while the viscous coefficient is hold constant to 1.5×10^{-4} m^2 s^{-1} over the whole water column. Deep convection is simulated through the enhancement of the vertical diffusivity to 1 m^2 s^{-1} in regions where the stratification becomes unstable. The bottom topography is derived from the $\frac{1}{12}^\circ$ ETOPO5 database (US National Geophysical Data Center) with some modifications made for isolated grid points, islands and straits; in particular the Strait of Gibraltar has been modified to be represented by two grid points in latitude. To adequately resolve the dynamics of the Modified Atlantic Water (MAW) and the Levantine Intermediate Water (LIW), the 42 vertical Z-levels are distributed with a resolution varying from 10 m at the surface up to 300 m in the deepest part of the basin, with an intermediate resolution of about 40-50 m between the depths 200-700 m (LIW depth range). Finally, the time step used for CGM is 1800 seconds.

The horizontal refinement ratio of the nested model is 3:1, thus the FGM horizontal model domain has a resolution of $\frac{1}{24}^\circ \times \frac{1}{24}^\circ$, while no refinement in the vertical has been used, i.e. the FGM uses the same vertical levels of the CGM and the same vertical diffusivity and viscosity parameterization. Due to the refinement ratio used, the time step adopted for FGM is 600 seconds. Horizontal viscous and diffusive terms used in the FGM are modeled with a bi-harmonic formulation with diffusivity and viscosity coefficients equal to 1.2×10^9 m^4 s^{-1} in the interior of the domain, while they are linearly increased toward the CGM values in the last 10 grid points in longitude on both open boundaries. Model topography has been obtained merging the ETOPO2 bathymetry with the very high resolution digitalized bathymetry provided by the Physical Oceanographic group of the University of Malaga.

III. Two-way grid-refinement technique

The two-way nesting procedure applied in this work is essencially the same used by [14].

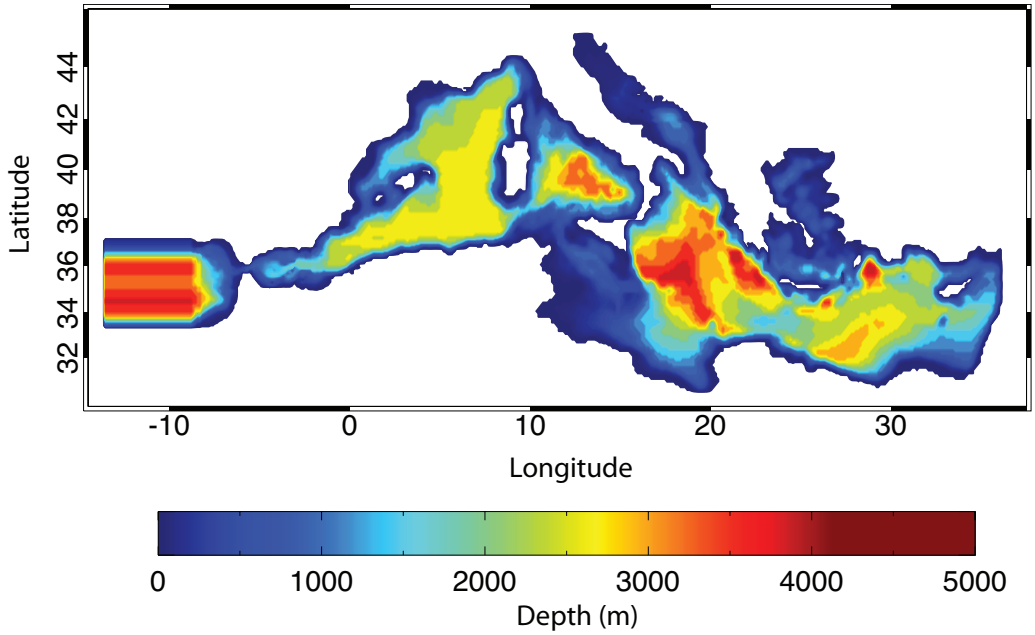


Fig. 2. Coarse grid ($\frac{1}{8}^\circ \times \frac{1}{8}^\circ$) model topography.

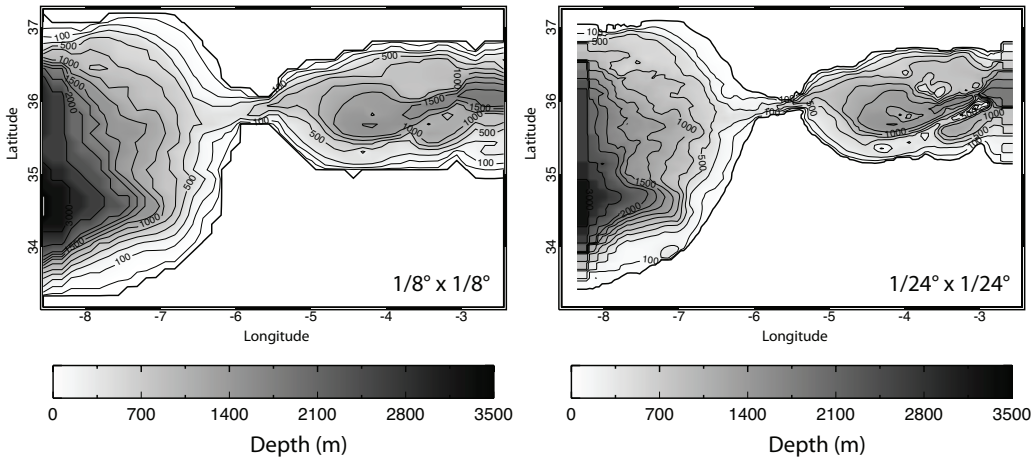


Fig. 3. (Left) Detail of the coarse grid model topography for the Alboran Sea and the Gulf of Cadiz. (Right) Fine grid ($\frac{1}{24}^\circ \times \frac{1}{24}^\circ$) model topography.

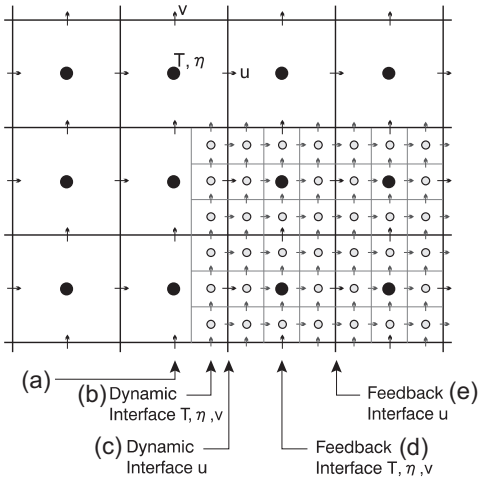


Fig. 4. Detail of the computational grid for the overlapping region of the FGM (gray) and the CGM (black). Circles show the position of scalar variables, while horizontal and vertical arrows represent the zonal and meridional velocity components respectively. The large black symbols are associated to the CGM, while the small gray symbols are associated to the FGM. Dynamical and feedback regions are also evidenced.Scalars and the tangent (to the nesting boundary) velocity component for the FGC are obtained interpolating the CGM values from columns **a** and **d** and imposed in column **b**. The normal velocity component is obtained interpolating the CGM values laying on the column **c** and imposed in the same column.The averaged FGM values of the scalars and the tangent velocities are replaced in CGM starting from column **d**, while for the normal velocity the feedback region begins with column **d**.

This approach can be summarized as follows: the FGM is driven at the open boundaries by interpolated CGM fields, while dynamical fields computed by the FGM are volume averaged and replaced in the CGM in the region covered by the FGM (Figure 4).

More in details the procedure followed can be schematized as:

1) the CGM is forwarded one time step;
2) the new CGM dynamical fields are interpolated on the dynamic interfaces (see Figure 4) and applied as lateral boundary conditions to the FGM. Due to the way the fine grid and coarse grid are staggered, the velocity component normal to the boundary only needs an interpolation tangent to the boundary (column **c** in Figure 4). To ensure the volume flux conservation between CGM and FGM, the interpolation is performed using the local second order interpolation scheme developed by [15]. This scheme uses three contiguous coarse data points to infer data in the fine cells covering the central coarse cell, moreover the mean value of the interpolated data on the fine cells corresponding to a coarse cell, is equal to the data in this coarse cell. Lateral boundary condition for the velocity component tangent to the boundary is applied in columns **b**. It is obtained by a bi-linear interpolation of the four nearest tangent velocity components of the coarse grid (columns **a** and **d**). The boundary condition for the scalars is obtained in two steps: during the first step they are linearly interpolated normally to the boundary (from columns **a** and **d** to column **b**) then, these new values are interpolated tangentially and applied again in column **b**;
3) the FGM is forwarded three time steps holding constant the lateral boundary conditions;
4) the new FGM dynamical values onto the overlapping CGM points, which are those points lying on, and inside the feedback interface (see Figure 4), are volume averaged and replaced into the CGM.

In the present nested model CGM and FGM are both treated as independent models, except for the communication parts in which the two models exchange each variable for interpolation and feedback. The two models run in parallel, together with an external driver that establishes the connection between CGM and FGM, and at the same time performs all the proper interpolations. The MPI message-passing standard is used for communication between the driver and the two models. Moreover, since both CGM and FGM are parallelized via horizontal domain decomposition, the exchange between those models and the single CPU dedicated to

the driver is performed in parallel. In other words each model sub-domain lying on, and inside the feedback interface, contemporarily send its data to the single CPU driver. In general these times are very small (0.1%) compared to the time necessary for the nested model to perform one time step. A similar small fraction of time is also needed to the driver to send back interpolated fields to FGM. On the contrary, the time spent by the driver to send back the interpolated 3D fields to CGM is not negligible as is evident in Table I. Here is shown the total time of a single step of the driver (T_{Dr}), that actually represents the total time of a single time step of the nested model, together with the time needed for the driver to send information back to CGM (T_{sC}). As expected, due to the domain decomposition technique used, the total time T_{Dr} progressively decreases as the number of CPUs dedicated to CGM and/or FGC increases, however, also T_{sC} increases as the number of CPUs dedicated to CGM increases. This behavior is not surprising because of the serial exchange of data established from the single CPU of the driver and the multiple CPUs of CGM. The impact of T_{sC} on the total nesting model time step, is represented as percentage ($\frac{T_{sC}}{T_{Dr}}$) in the last column of Table I. As can be seen, this percentage progressively increases as the total number of CPUs increases, reaching a value of more than 40% when a total number of 61 CPUs (40 for CGM and 20 for FGM) is used.

Number of Processors			Time		
Driver	CGM	FGM	T_{sC}	T_{Dr}	%
1	4	2	281	4015	7.0
1	8	2	366	3128	11.7
1	16	2	463	2776	16.7
1	32	2	623	2745	22.7
1	32	4	623	2136	29.2
1	32	12	623	1707	36.5
1	40	12	680	1681	40.4
1	40	15	680	1647	41.3
1	40	20	680	1620	42.0

TABLE I
ELAPSED TIME (IN MILLISECONDS) NEEDED FOR THE DRIVER TO SEND INFORMATION TO THE CGM MODEL (T_{sC}), THE TOTAL TIME FOR A SINGLE STEP OF THE DRIVER (T_{Dr}), I.E. THE TOTAL TIME FOR A SINGLE TIME STEP OF THE NESTED MODEL, AND THE PERCENTAGE $\frac{T_{sC}}{T_{Dr}}$. THE EVALUATION IS PERFORMED FOR DIFFERENT CPUS DISTRIBUTIONS

Number of Processors			Time			
Driver	CGM	FGM	T_{rC}	T_{rF}	T_{sC}	T_{Dr}
1	4	2	2067	1478	281	4015
2	4	2	2067	1478	140	3874
4	4	2	2067	1478	72	3806
1	8	2	1095	1478	366	3128
2	8	2	1095	1478	185	2947
4	8	2	1095	1478	93	2855
1	16	2	646	1478	463	2776
2	16	2	646	1478	232	2545
4	16	2	646	1478	118	2431
1	32	2	455	1478	623	2745
2	32	2	455	1478	313	2425
4	32	2	455	1478	109	2231
1	32	4	455	869	623	2136
2	32	4	455	869	313	1826
4	32	4	455	869	109	1622
1	32	12	455	440	623	1707
2	32	12	455	440	313	1397
4	32	12	455	440	109	1193
1	40	12	372	440	680	1681
2	40	12	372	440	341	1342
4	40	12	372	440	173	1174
1	40	15	372	406	680	1647
2	40	15	372	406	341	1308
4	40	15	372	406	173	1140
1	40	20	372	379	680	1620
2	40	20	372	379	341	1281
4	40	20	372	379	173	1113

TABLE II
ELAPSED TIME (IN MILLISECONDS) INDICATING A SINGLE CGM TIME STEP (T_{rC}), A SINGLE FGM TIME STEP (T_{rF}), THE TIME NEEDED FOR THE DRIVER TO SEND INFORMATION TO CGM (T_{sC}), AND THE TOTAL TIME FOR A SINGLE STEP OF THE DRIVER (T_{Dr}), I.E. THE TOTAL TIME FOR A SINGLE TIME STEP OF THE NESTED MODEL. THE EVALUATION IS PERFORMED FOR DIFFERENT CPUS DISTRIBUTIONS

It appears evident that the serial driver works as a bottle neck for the entire nesting system.

Thus, in order to strongly reduce T_{sC}, the driver has been parallelized. In this way the sending procedure from the driver to CGM is performed contemporary by more CPUs resulting in a total reduction of T_{sC}. As can be seen in Table II, the reduction scales quite linearly with the number of CPUs dedicated to the driver. As an example the reduction obtained when 4 CPUs are used for the driver instead of 1, in the last configuration of Table II, i.e. 40 CPUs for CGM and 20 for FGM, is about 31%.

IV. Model Results

The model is initialized with MEDATLAS-II ([16]) climatology data for the month of August. A 3D relaxation of salinity and temperature toward the climatological monthly Levitus data ([17]) is applied in the first 10 grid points in longitude of CGM, in the so called Atlantic box, with a relaxation time of 5 days. The model is forced at the surface through the specification of the wind stress and heat fluxes taken from the ERA40 reanalysis data base provided by the European Centre for Medium-Range Weather Forecasts (ECMWF). The reanalysis used covers the 1958-1997 period, has a time frequency of six hours, and a spatial resolution of about $1.125° \times 1.125°$. As recently demonstrated by [18] this low resolution has the effect of underestimating the total winter heat loss over the area of the Gulf of Lion (hereinafter GoL) inducing a too shallow deep convection in that area. To overcome this problem the winter (DJF period) heat fluxes over the GoL area have been corrected following the same technique used recently by [19]. Freshwater fluxes (evaporation, precipitation, and river runoffs) are applied as a virtual salt flux through a relaxation of the modeled sea surface salinity (SSS) toward the monthly SSS of MEDATLAS with a relaxation time of 5 days.

The model simulates two ERA40 loops, the first 40 years are considered as spin-up, thus we will show results only from the second 40 years. In the rest of the paper the term "time-mean" will be used to indicate the time average over this second 40 years. The entire simulation has been integrated on the CRESCO (Centro di computazionale di RicErca sui Sistemi Complessi) system present in ENEA.

	°C or psu	SST	$\langle T \rangle$	SSS	$\langle S \rangle$
Mediterranean	OBS	19.61	13.74	38.22	38.61
Sea	MDL	18.91	13.72	38.16	38.60
Gulf of	OBS	17.71	13.02	37.97	38.42
Lion	MDL	17.35	12.82	37.91	38.34
Levantine	OBS	20.73	13.96	39.07	38.78
Basin	MDL	19.24	14.02	39.01	38.80
Adriatic	OBS	17.76	13.99	38.20	38.59
Sea	MDL	17.66	13.66	38.17	38.46

TABLE III
MODEL (MDL) SST (IN °C), 3D AVERAGED TEMPERATURE ($\langle T \rangle$ IN °C), SSS (IN PSU) AND 3D AVERAGED SALINITY ($\langle S \rangle$ IN PSU) AVERAGED OVER THE 1958-1997 PERIOD. THE EQUIVALENT VALUE FOR THE MEDATLAS-II DATABASE ARE INDICATED AS OBS.

The resulting time-mean Mediterranean surface heat loss is equal to 5.5 W m^{-2} with an inter-annual standard deviation of 3.6 W m^{-2}. This value is in good agreement with previous direct and derived measurements ([20], [21], and [22]). The time-mean fresh water flux (Evaporation-Precipitation-River run-off) derived by the SSS relaxation over the entire Mediterranean basin is 0.72 ± 0.04 m $year^{-1}$. This value is in good agreement with the one obtained by [23] (0.71 ± 0.07 m $year^{-1}$), and, in general, it is in the observed range estimated in previous works (e.g. 0.95 m $year^{-1}$ in [20], 0.52 m $year^{-1}$ in [24]).

The time-mean volume transport simulated by the model through the Strait of Gibraltar is 1.03 ± 0.04 Sv. This value belongs to the range of previous estimations ($+0.72 \pm 0.16$ Sv, [25]; 1.68 Sv, [20]; 0.77 Sv, [26]; 0.81 ± 0.07 Sv, [27]).

c In Table III the time-mean temperature and salinity simulated by the model are compared with MEDATLAS-II climatological values. For the sea surface temperature (SST) the model shows for the whole basin a cooling bias of $-0.70°$C. More in detail a maximum cooling bias of $-1.49°$C is present in the Levatine basin, while a better agreement is found

for the Adriatic Sea ($-0.10°$C) and the GoL ($-0.36°$C). Because of the surface salinity relaxation applied in the model, a very good agreement has been found for the SSS, with differences that do not exceed 0.06 psu. A good agreement is found for the 3D averaged temperature and salinity fields ($\langle T \rangle$) and ($\langle S \rangle$) computed following the equation:

$$\langle Trac \rangle = \frac{\iiint_V Trac\,(x,y,z)\,dxdydz}{\iiint_V dxdydz}, \quad (1)$$

where $Trac$ represents temperature or salinity and V is the volume of the generic basin. Over the whole basin the difference in temperature is only $-0.02°$C. The most biased basins are the Adriatic Sea, with a bias of $-0.33°$C, and the GoL with a bias of $-0.20°$C, while the Levantine basin is characterized by a lower bias of +0.06°. For the salinity the maximum bias occurs in the Adriatic Sea with a fresh bias of -0.13 psu, and the GoL with a fresh bias of -0.08 psu, while for the Levantine basin the positive bias is limited to 0.02 psu.

V. Conclusion

In this work has been developed a two-way grid-refinement procedure that is both simple to implement and efficient in terms of computational time. The procedure has been developed for the MITgcm model even if it could be applied in principle to any Arakawa C-grid ocean model. The grid-refinement method used was basically the same described by [14]. Technically the refined grid belongs to a separate model that interactively exchange information with the eddy-permitting model via an external driver that performs, also, all the necessary interpolations. This driver represents the main core of the developed procedure. It has been shown that to strongly reduce the latency between the driver and CGM, i.e. the time required for the message-passing process between the driver and CGM, the driver needed to be parallelized. Doing so a very good performance has been obtained.

The nested model has been used to simulate the present climate of the Mediterranean Sea. The two-way grid refinement has been applied at the Strait of Gibraltar. The model has been forced by the six-hourly ERA40 heat flux and wind stress for the period 1958-1997, while the surface freshwater fluxes have been modeled through a surface salinity relaxation toward climatological monthly data. The model has been integrated stably for a 80-year period (2 ERA40 loops). It has been shown that the model is able to reproduce most of the main characteristics of the Mediterranean circulation, moreover we stress that the transition of the dynamical fields between FGM and CGM at the nesting boundaries appeared smooth during the entire simulation confirming the robustness of the adopted grid-refinement procedure.

Acknowledgment

We would like to thank the people working in the ENEA center of Portici who have strongly supported the implementation of our nested model on the CRESCO system.

References

[1] H. Bryden and H. Stommel, "Limiting processes that determine basic features of the circulation in the mediterranean sea," *Oceanol. Acta*, vol. 7 (3), pp. 289–296, 1984.

[2] G. Sannino, L. Pratt, and A. Carillo, "Hydraulic criticality of the exchange flow through the strait of gibraltar," *Submitted to J. Phys. Ocean.*, 2009.

[3] V. Roussenov, E. Stanev, V. Artale, and N. Pinardi, "A Seasonal model of the Mediterranean Sea General Circulation." *Journal Of Geophysical Research-Oceansc*, vol. 100, no. C7, pp. 13 515–13 538, 1995.

[4] M. Zavatarelli and G. Mellor, "A Numerical Study Of The Mediterranean-Sea Circulation," *Journal Of Physical Oceanography*, vol. 25, no. 6, Part 2, pp. 1384–1414, 1995.

[5] P. Wu and K. Haines, "The general circulation of the Mediterranean Sea from a 100-year simulation," *Journal Of Geophysical Research-Oceans*, vol. 103, no. C1, pp. 1121–1135, 1998.

[6] V. Artale, D. Iudicone, R. Santoleri, V. Rupolo, S. Marullo, and F. D'Ortenzio, "Role of surface fluxes in ocean general circulation models using satellite sea surface temperature: Validation of and sensitivity to the forcing frequency of the Mediterranean thermohaline circulation," *Journal Of Geophysical Research-Oceans*, vol. 107, no. C8, 2002.

[7] M. Tonani, N. Pinardi, S. Dobricic, I. Pujol, and C. Fratianni, "A high-resolution free-surface model of the Mediterranean Sea," *OCEAN SCIENCE*, vol. 4, no. 1, pp. 1–14, 2008.

[8] A. Bozec, P. Bouruet-Aubertot, K. Beranger, and M. Crepon, "Mediterranean oceanic response to the interannual variability of a high-resolution atmospheric forcing: A focus on the Aegean Sea," *Journal Of Geophysical Research-Oceans*, vol. 111, no. C11, NOV 25 2006.

[9] E. Demirov and N. Pinardi, "Simulation of the Mediterranean Sea circulation from 1979 to 1993: Part I. The interannual variability," *Journal Of Marine Systems*, vol. 33, pp. 23–50, 2002.

[10] G. Korres, N. Pinardi, and A. Lascaratos, "The ocean response to low-frequency interannual atmospheric variability in the Mediterranean sea. Part I: Sensitivity experiments and energy analysis," *Journal Of Climate*, vol. 13, no. 4, pp. 705–731, 2000.

[11] J. Marshall, A. Adcroft, C. Hill, L. Perelman, and C. Heisey, "A finite-volume, incompressible navier stokes model for, studies of the ocean on parallel computers," *Journal of Geophysical Research C: Oceans*, vol. 102, no. C3, pp. 5753–5766, 1997.

[12] J. Marshall, C. Hill, L. Perelman, and A. Adcroft, "Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling," *Journal of Geophysical Research C: Oceans*, vol. 102, no. C3, pp. 5733–5752, 1997.

[13] A. Adcroft, C. Hill, and A. J. Marshall, "Representation of topography by shaved cells in a height coordinate ocean model," *Monthly Weather Review*, vol. 125, no. 9, pp. 2293–2315, 1997.

[14] A. D. Fox and M. Maskell, "Baroclinic response of the Adriatic Sea to an episode of bora wind," *Journal of Physical Oceanography*, vol. 26, no. 7, pp. 1354–1369, 1996.

[15] P. Nomérange, "Implementation of a nesting procedure in the gher 3d model. application in the gulf of cadiz." 3^{rd} MMARIE annual meeting, 1998.

[16] MEDAR Group, "Medatlas/2002 database. mediterranean and black sea database of temperature salinity and bio-chemical parameters. climatological atlas." *IFREMER Edition.*, 2002.

[17] S. Levitus, *Climatological Atlas of the World Ocean*, 1982.

[18] M. Herrmann and S. Somot, "Relevance of ERA40 dynamical downscaling for modeling deep convection in the Mediterranean Sea," *Geophys. Res. Lett.*, vol. 35, L04607, 2008.

[19] M. Herrmann, S. Somot, F. Sevault, C. Estournel, and M. Déqué, "Modeling the deep convection in the Northwestern Mediterranean sea using an eddy-permitting and an eddy-resolving model: case study of winter 1986-87," *J. Geophys. Res.*, vol. 113, C04011, 2008.

[20] J. Bethoux, "Budgets of the Mediterranean sea. Their dependence on the local climate and on the characteristics of the Atlantic waters." *Oceanol. Acta*, vol. 2 (2), pp. 157–163, 1979.

[21] Bunker, Charnock, and Goldsmith, "A note on the heat balance of the Mediterranean and Red Seas," *J. Mar. Syst.*, vol. 40, pp. 73–84, 1982, supplement.

[22] A. Macdonald, J. Candela, and H. Bryden, "An estimate of the net heat transport through the Strait of Gibraltar," in *Seasonal and Interannual Variability of the Western Mediterranean Sea*, ser. Coastal Estuarine Studies 46., P. L. V. Ed., Ed. Amer. Geophys. Union, 1994, pp. 13–32.

[23] C. Gilman and C. Garrett, "Heat flux parameterizations for the mediterranean sea: the role of atmospheric aerosols and constraints from the water budget," *Journal of Geophysical Research*, vol. 99, no. C3, pp. 5119–5134, 1994.

[24] C. Garrett, " The role of the strait of Gibraltar in the evolution of Mediterranean water, properties and circulation." *Bull. Inst. Océanogr.*, vol. Monaco, no. 17, pp. 1–19, 1996.

[25] H. Bryden, J. Candela, and T. Kinder, "Exchange through the strait of Gibraltar," *Prog. Oceanogr.*, vol. 33, pp. 201–248, 1994.

[26] M. Boukthir and B. Barnier, "Seasonal and interannual variations in the surface freshwater flux in the Mediterranean Sea from the ECMWF re-analysis project," *J. Mar. Syst.*, vol. 24, pp. 343–354, 2000.

[27] B. Baschek, U. Send, J. G. Lafuente, and J. Candela, "Transport estimates in the Strait of Gibraltar with a tidal inverse model," *J. Geophys. Res.*, vol. 106(C12), pp. 31,03331,044, 2001.

The TEPHRA Model: Grid Implementation on PI2S2 Project

Reitano D.[1,2], Coltelli M. [1], D'Agostino M.[1,2], Drago A.[2], Lizzio S.[2], Pistagna F.[2], Prestifilippo M.[1], Scollo S.[1], Spata G.[1], Reitano D.[1,2], Cariolo S.[2], Russo G.[2]

[1]INGV – Istituto Nazionale di Geofisica e Vulcanologia, Piazza Roma, 2 95125 Catania, Italy, reitano@ct.ingv.it
[2]Consorzio COMETA, via Santa Sofia, 64 – 95100 Catania, Italy

***Abstract* - The "TEPHRA" model describes the dispersal and sedimentation processes of volcanic particles produced during explosive eruptions. The model is used by the Istituto Nazionale di Geofisica e Vulcanologia, sezione di Catania (INGV-CT) to forecast tephra dispersal and produce maps of tephra fallout of two eruptive scenarios at Mt. Etna. These maps are then transferred to Italian Civil Protection, which manages the public warnings and plans mitigation measures. The TEPHRA model is implemented in ANSI-C code and uses MPI commands to maximize the parallel computation. In order to provide a better performance, we worked on porting the model into the PI2S2 Sicilian Grid infrastructure. We configured the application to run it on the Grid, using gLite middleware, we also analyzed its performances and compared them with these ones obtained on the cluster at INGV-CT, where the simulations are run daily. Finally, this work reports the advantages obtained introducing a customized high priority queue, called *Emergency Queue,* to resolve the job-scheduling time delay.**

Index terms: COMETA, ash fallout, early warning

I. INTRODUCTION

THE TEPHRA model is used by INGV-CT, to forecast tephra deposits produced during Etna explosive eruptions. The model is implemented in ANSI-C, and uses the MPI (Message Parsing Interface) libraries for running in distributed computation. In this way several simulations are performed in a very brief time. The model is consequently used to compile probability maps and hazard curves where a great number of simulations are needed. TEPHRA was used to construct the hazard assessment at Tarawera, in New Zealand (Bonadonna et al, 2005) and at Mt. Etna, in Italy (Scollo, 2006). Actually, the model runs on a Beowulf cluster at INGV-CT data center for simulating two different eruptive scenarios: the 1998 eruption (Andronico et al., 1999) in which an eruptive column rose up to 12 km of height (a.s.l.), and the 2002-03 Etna eruption (Andronico et al., 2008) in which the column reached 7 km (a.s.l) and lasted several days (Figure 1). Goal of this work is to test and evaluate the model performances into a new Grid infrastructure, inside the so called *PI2S2 project* (Progetto per l'Implementazione e lo Sviluppo di una e-Infrastruttura in Sicilia basata sul paradigma Grid – www.pi2s2.it*)*, managed by the Consorzio Cometa (www.consorzio-

cometa.it).
The porting activity was the first step inside this project. All results have been analyzed and compared with the performances obtained by the cluster a INGV-CT.

Fig.1 Eruption column during 2002-03 Etna eruption. Photo courtesy of INGV-CT

II. HAZARD ASSOCIATE TO THE ASH FALLOUT

Volcanic ash (volcanic particles having diameter < 2 mm) erupted during explosive eruption may be transported in the atmosphere for days to months due to their fine dimensions and travel for a long distance far from the volcanic vent where they were produced. Volcanic ash can represent a great hazard for aviation safety and population. In fact, it is very dangerous to the air traffic (Casedevall, 1994). In fact, it melts in the combustion chamber of the aircraft. Its subsequent solidification on pallets and moving parts of the turbines would compromise their operation causing serious turbine damages. Volcanic ash is also being very hard and abrasive and can erode the structure of the aircraft and flight surfaces. In particular, the abrasion of the glass cockpit can reduce the long-range vision of the pilot. Because weather radar aboard the aircraft can fail to detect volcanic ash, the prediction and early warning of their presence in the atmosphere, especially at night and in low visibility condition are still the only real way to prevent disasters. Moreover, tephra fallout can give problems for people living around a densely inhabited volcano as Mt. Etna. In particular, tephra fallout may affect the transport system, causes damages to vegetation, to water supply and to airport operations.

Fig.2 Ash fallout on Catania Airport

A. The TEPHRA model

TEPHRA is an advection and diffusion model, which describes the diffusion, transport and sedimentation of volcanic particles. The model solves semi-analytically the mass conservation equation (Bonadonna et al., 2005) including grain-size dependent diffusion law (Suzuki et al., 1983), particle density variation, stratified atmosphere (Macedonio et al., 1988), terminal settling velocity function of the particle Reynolds' Number (Bonadonna et al., 1998). The choice of the model was strictly linked to the parallel computing technique available in TEPHRA (it is written in ANSI C and uses MPI commands). In fact, TEPHRA makes easy the use of probabilistic approaches that requires a great number of simulations. In a typical simulation, the total space around the summit craters is divided in many points of a

Grid in which the total mass falling on the ground is evaluated: an approximation of 5000 points produces satisfactory maps of the tephra fallout (Figure 3).

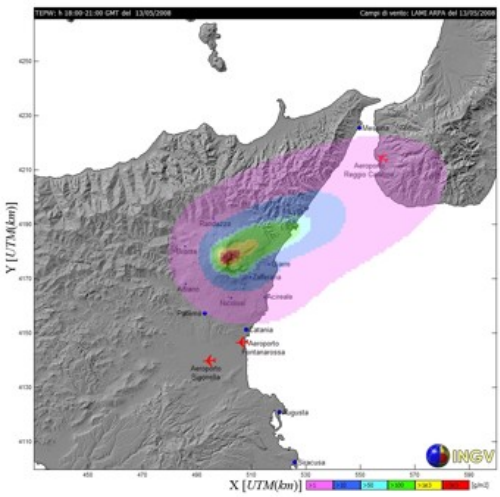


Fig.3 Map of tephra fallout obtained by the TEPHRA model.

The time computing is reduced using the parallel calculation. The parallel model used by TEPHRA is the SPMD (Single Program - Multiple Data). Multiple instances of the same code are executed in parallel on different nodes (cores). This is possible because the results obtained from the model on a single Grid point do not depend on others. So, the entire set of data input is split between different nodes. Each node performs the real computation of the model with its own set of partial input. At the end a master node is responsible to collect and sort the results.
The inputs of the TEPHRA model are:

- a model configuration file;
- parameters that identify the geographical area;
- volcanological parameters function of the simulated eruption;
- meteorological data (direction and speed of winds).

B. The algorithm used by INGV Catania

The cluster present at INGV-CT (Figure 4), where the simulations are performed, consists of 24 different servers. They are HP Proliant DL 145 Opteron 248. Each of these machines has two 2.6 GHz/1MB processors, RAM 1 GB and 80 GB Hard Disk Drive (HDD). Only the master node has an additional 160 GB HDD. The servers are connected via a HP ProCurve switch 2810-48g with 64 MB of SDRAM and 16 MB of flash memory. The switch has 48 1Gbps ports so it possible to use both Ethernet ports of each server. The installed operating system is SUSE Linux Enterprise Server and various compilers and libraries for C and Fortran are present, including the MPI libraries for parallel calculation.
Every day, by means of an automated system realized by INGV-CT, meteorological data are supplied by the Italian Air Force (CNMCA) and by Civil Protection (ARPA-SIM). Using forecasting wind data, the model simulates two different eruptive scenarios, corresponding to the 1998, 2002-03 Etna eruption, whose deposits have been already analyzed in detail (Andronico et al., 1999; Andronico et al., 2008, respectively).

Fig.4 Cluster at INGV – CT

The meteorological data are in *grib* format. By using scripts files, implemented by INGV-CT, parameters like the intensity and direction of winds, the temperature, the pressure and the humidity are extracted and introduced into the model. TEPHRA is hence run producing a simulation every three hours for the next two days; the system produces, hence, 16 different simulations. These maps are supplied to the Civil Protection Department that will take care to warn the various authorities, launch alarms, take decisions and any necessary precautionary measures for the public safety (Figure 5).

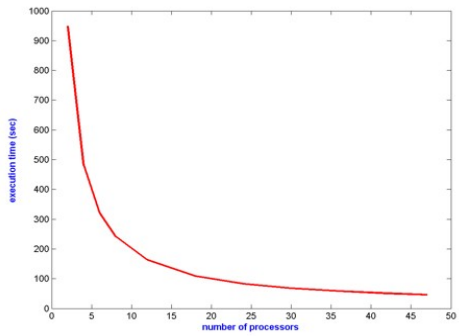


Fig.5 Execution time (in sec) obtained on cluster Vs core numbers at INGV-CT

C. The PI2S2 project

The INGV is a member of the consortium COMETA (***CO****nsorzio* ***M****ulti* ***E****nte per la promozione e l'adozione di* ***T****ecnologie di calcolo* ***A****vanzato*) that manages the PI2S2 project. This project, integrated in the Programma Operativo Nazionale (PON) *"Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" 2000-2006*, has proposed to provide the Sicily of a large distributed computing and storage infrastructure to serve the scientific communities and private businesses. This structure wants to be a Virtual Laboratory able to provide on demand services to Universities, Research Institutes and small and medium enterprises (SME).

Using the Grid performance the increase of the cooperation between scientific institutions, universities and modern enterprises is expected. The nodes of the Grid deployed for the project PI2S2 are located geographically between the three poles of Catania (INAF Laboratories, INFN and University of Catania), Palermo and Messina (University Campus). The Sicilian Grid is connected to the European Grid using the middleware (gLite on Unix-Linux platform).

After an initial phase of design, construction and implementation of computing through the installation of hardware and middleware, the project is now in its final phase, consisting of testing the software inside the middleware and evaluating the performance of Grid computing.
Inside the research activity of INGV-CT many scientific applications uses the Grid benefits and the knowledge base acquired inside the *PI2S2* project.

D. Porting Tephra on the Grid

TEPHRA is written in ANSI C language, and uses the GC libraries ver. 6.7 distributed with open source license from HP™ as an aid for the management of memory (i.e. malloc function) (www.hpl.hp.com/personal/Hans_Boehm/gc/). It uses Message Passing Interface (MPI) Ver.1, *de facto* standard for the creation of parallel code in Fortran or C, based on the exchange of messages between processors.
GCC, Intel and PGI C language compilers are present in all nodes of the PI2S2 Grid, and in particular in the User Interface, so they are directly available by all users according to a strict access control policy (Public Key Infrastructure type), based on a system of digital certificates and public key cryptography.
The porting phase on Grid was tested using different approaches, in order to achieve better the stability and efficiency during the runtime. TEPHRA was compiled through the User Interface and the system configuration is identical for all the Worker Nodes where the application is then executed.
Using Job Description Language (JDL) directives, a script file defines a set of parameters and configurations that are necessary for running the application. In particular, it specifies the input and output files, any requirements for the submission, the type of MPI implementation, and the requested number of nodes. In this way each application can run after the submission of a "job", properly described in this script file. Each job produces a single simulation of the model using different set of meteorological data.
In order to optimize the execution time, different strategies are tested for the job submission. We used customized Bash shell scripts to submit 16 different jobs simultaneously for all simulations.
Many tests have been conducted in order to complete the initial trial phase. Grid computation is very competitive than the simulations obtained by cluster at INGV–CT. Nevertheless, the most important problem found on Grid was due to scheduling time because, sometimes, it is too long and, overall, not predictable.

E. The Emergency Queue

The INGV-CT operates full time H24 in order to guarantee surveillance and monitoring activity and furnish all the significant information to the Civil Protection, when it is necessary.
During periods of normal activity, but especially in case of emergency, it is essential to have the results of simulations in a very short time. So redundancy, QoS and good system performance are necessary requirements if INGV-CT wants to use the Grid structure for giving speed information to the Civil Protection.
In order to solve this problem, with the help of PI2S2 system managers group and Grid experts, a high priority job was created. This service, called *Emergency Queue* is reserved only for particular users with appropriate permissions. It can temporarily stops all jobs running on a given site of the Grid to allow the high-priority job (i.e. to the TEPHRA model), bypassing all scheduling times due to the load of the other job queues at lower priority. The Emergency Queue system was tested first on a single site of the Grid, and

guaranteed a good performance for the TEPHRA model: the time waiting was reduced up to 50% in some cases. Now it is implemented on the whole infrastructure and it can be available for particular applications (Figure 6 and 7).

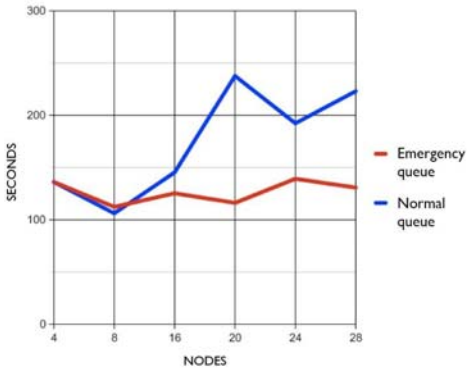


Fig.6 Plot of waiting time of a Tephra job on Grid varying the numbers of nodes. Emergency queue (in red) shows a marked improvement compared to the normal queue.

III. Conclusion

Numerous tests were performed, in order to evaluate the effective improvements due to Grid parallel computing, trying to achieve a measure of real performance. Moreover testing process permits to optimize the submission strategies.

We used the submission time for comparisons, taking one single job as a sample. The use of Bash scripts, at the beginning and at the end of the job execution, gives us the real execution time workload. Instead, the time evaluation of the overall process of the job submission is obtained using the specific functions of the Workload Management System, a suite of the gLite middleware for managing the job on Grid. As a difference, we have also calculated the whole scheduling time, the waiting time between the submission of the job and its correct execution.

Several submissions have been tested using different number of nodes, analyzing the average times obtained.

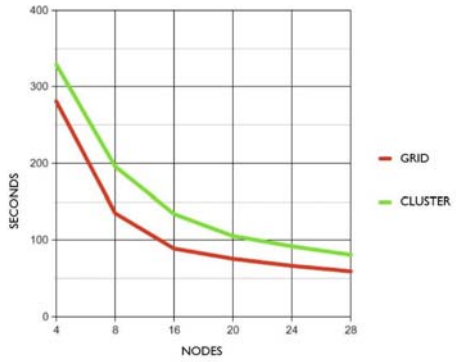


Fig.7 Simulation of the TEPHRA model running (in red) on the Emergency Queue of PI2S2 Grid network, compared with the same number of nodes on the INGV local cluster (in green).

As expected, the execution time decreases if the number of nodes increases. However, the improvement of execution time is insignificant if over 20 nodes are used.

Comparing this workload with the same simulations running on the cluster at INGV-CT, we obtained an advantage up to 26% in terms of execution time.

Acknowledgments

The authors would like to thank the UFSO staff.

References

Andronico, D., Del P. Carlo, and M. Coltelli (1999), The 22 July 1998 fire fountain episode at Voragine Crater (Mt Etna, Italy), Volcanic and Magmatic Studies Group, Annual Meeting, Birmingham, 5 – 6 January.

Andronico, D., Scollo, S., Caruso, S., Cristaldi, A., "Tephra dispersal and features of deposit", JGR, 2008, 113,B04209.

Bonadonna, C., Ernst, G.G.J., Sparks, R.S.J., 1998. Thickness variations and volume estimates of tephra fall

deposits: the importance of particle Reynolds number. J.Volcanol.Geotherm. Res. 81, 173–187

Bonadonna, C., Phillips, J.C., Houghton, B.F., "Modeling tephra sedimentation from a Ruapehu weak plume eruption, JGR, 110, B08209, doi:10.1029/2004JB003515.

Casadevall, T. J. (1994), Volcanic ash and aviation safety, Proceedings of the First International Symposium on Volcanic Ash and Aviation Safety, U.S. Geological Survey Bulletin 2047, 450 pp.

Macedonio, G., M. T. Pareschi, and R. Santacroce (1988), A numerical simulation of the Plinian fall phase of 79 AD eruption of Vesuvius, J. Geophys. Res., 93, 14,817– 14,827.Scollo, S.2006 phd.

Scollo, S. (2006), Fallout processes of volcanic particles from Etna explosive eruptions, PhD thesis, pp. 158–201, Univ. of Bologna, 28 June.

Monitoring and Analyzing of Volcanoes Activity by using INGV Cameras and Infrasound Sensors Network

Reitano D.[1,2], Cariolo S.[2], Russo G.[2]
[1]*INGV – Istituto Nazionale di Geofisica e Vulcanologia, Piazza Roma, 2 95125 Catania, Italy, reitano@ct.ingv.it*
[2]*Consorzio COMETA, via Santa Sofia, 64 – 95100 Catania, Italy*

Abstract

The main scope of this work is to compare two different techniques used to analyze volcano activity, inside PI2S2 grid network. The first aim is to monitor strombolian activity in order to automatically classify different types of explosions and achieve real time identification of critical ones using infrared thermal camera images. The application runs both on a local PC and on the grid network. Referring to the second part discussed on this work, the goal is to make clustering of explosion recorded by infrasound sensors, placed on Mt. Etna volcano.

Index terms: COMETA, pattern recognition, thermal images, infrasound signals.

I. INTRODUCTION

THE Istituto Nazionale di Geofisica e Vulcanologia, Catania branch, works on geo-science researches by monitoring and analyzing data recorded from different kind of sensors, sending them, in real time, to the Control Room placed in the Catania Centre. In recent times, multidisciplinary observation of active volcanoes, using technological instrumentation, represents a very useful technique to better evaluate significant changes in volcanoes behaviour. In particular we are going to introduce cameras and infrasound sensors works based on.

Over the last 20 years, INGV has been involved in planning the use of video cameras for monitoring and surveillance activity. Thirteen different video stations operates today on active volcanoes. The network is composed by visible, infrared (IR) and thermal cameras. All video signals are recorded continuously and sent, in real time, to the Control Room in Catania.

The infrasound signals seems to be very useful to understand volcanic activity if they are investigated in a multidisciplinary approach (i.e. with thermal images). The proposed algorithm gets the events by using a method based on standard deviation squared series (SD). By using two different thresholds on the SD series, it is possible to recognize different events together with their starting and ending points. Moreover, using a cross correlation algorithm (Green and Nueberg, 2006), the system is able to classify each event.

Under the PI2S2 project we try to conduct a study regarding the benefits, in terms of performance, resulting from the execution of scientific applications on Grid infrastructure rather than on individual PCs or on a local cluster. In particular in this work we present the results of a comparative analysis of performance on a single PC and on GRID

infrastructure of the two different scientific applications described above, analyzing seismic and volcanic signals. Several test are conducted in order to find the better performances of both algorithms.

II. The Presa Application

The application called PRESA (**P**attern **R**ecognition to monitor **S**trombolian **E**xplosive **A**ctivity), was born in order to trigger and classify the most significant explosions recorded on the summit craters of Mt. Stromboli volcano. It uses, as input data, the frames recorded by a video camera operating in the infrared (IR) spectrum placed on this volcano. The results could be very useful for monitoring and surveillance purposes, and for statistical analysis of the recorded data.

A. Description of the algorithm

The application consists of several modules:

- The *Loader* module loads the images to be analyzed, starting from video frames or real time video stream;
- The *Trigger* module analyzes the acquired images and select those in which the system recognize an event;
- The *Scanner* module which classifies the events found by the Trigger, rejecting any false alarms.

So the proposed system can be divided into two phases: the recognition and the classification of explosive phenomena. The trigger module performs a binarization of the images acquired by an IMAQ video card (provided by National Instruments™) and recognizes an explosion if certain condition are reached:

- There are a minimum number of white pixel in the scanned area;

- At least 10 frames have got this feature

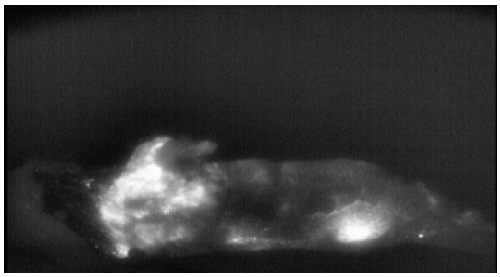


Fig.1 An example of an infrared image of summit craters on Mt. Stromboli

Fig.2 The three triggered areas, recognized in Fig.1 and added into the same frame

The next step is the classification of the images, therefore the analysis of the highlighted regions. Various techniques have been tested in order to make a meaningful classification of the explosion. The problem is to find the geometric figure that could better represent the area of interest. Starting from the bounding box, the rectangle with minimum area that contains the entire explosion, we try also the elliptical shape and at last the convex hull.

Fig.3 The convex hull region

B. Benchmarks on a local Pc and on the PI2S2 Grid network

Several tests have been performed on the PI2S2 Grid and on a local PC where the application has been developed.
In particular on the grid we adapted the code for different submission:

- Single job
- Collection job
- DAG job

First, a single job was tested without using parallel code; then the code was divided into three independent modules (a collection job) and test on the grid. Last test was the Direct Acyclic Graph (DAG) a directional graph, with no cycles, in which each vertex consists of a job on a particular task of the application.

C. Results

The trigger and classification modules are tested continuously with real time images on a local Pc.
The results are very satisfactory; the test pattern has been realized using 54 explosions, 42 ash emissions and 6 non volcanic phenomena. The obtained percentages are summarized in Table 1

Explosions	Ash emissions	Non volcanic phenomena
54	42	6
100%	66%	100%

Table 1. Patterns used for the tests

Execution time on grid is satisfactory, but due to scheduling time, overall time sometimes is not competitive.

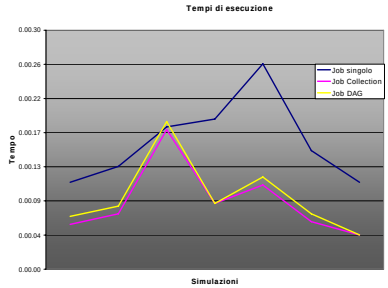


Fig.4 Execution time of the Presa model

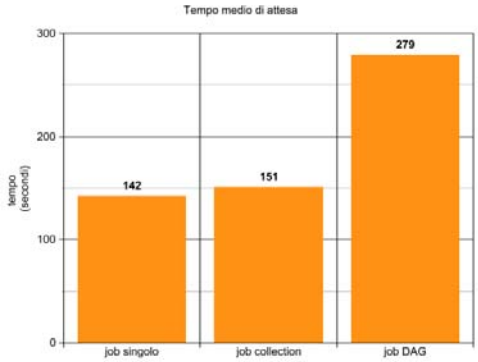


Fig.5 Average waiting time

III. THE INFRASOUND ALGORITHM

A. The infrasound signals

Since 2006, the INGV Catania, has realized a new network based on infrasound stations. The network is located in the summit area of Mt. Etna, where analog signals are sampled

in real time and sent to the Control Room inside the seismic dataset. Infrasound signals are emitted during eruptions from active volcanoes and since last years, some INGV researchers, using modern technologies, are investigating this field. The explosive volcanic activity, in fact, is always accompanied by changes in acoustic waves below 20 Hz, which are impulsive transient in the infrasound spectrum (operative range: 0 - 7Hz).

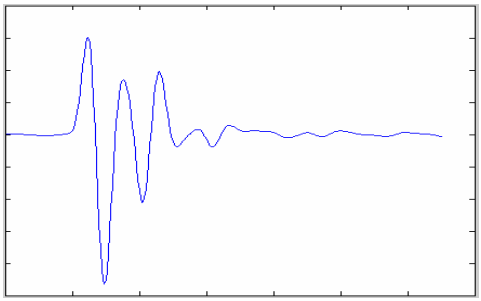


Fig. 6 Example of an infrasound event

Each acquired event, contains information on the geophysical phenomena and, in particular, describes the activity of the different Mt. Etna craters. In the present work a new automated methodology for the classification of infrasound events is proposed. Goal of the work is to classify infrasound events in groups with common characteristics, and to find some features that best describe these families of events. This was carried out using computational tools and advanced techniques for High Performance Computing (HPC). In particular we use distributed computing (GRID), made available by the PI2S2 project (PON 2000 – 2006) managed by the Consorzio COMETA to which INGV belongs. Different working environments have been used, in particular Matlab™ for Windows for the manipulation of different infrasound dataset, and the C language on OS Linux for the implementation of genetic algorithm and for distributed computation.

B. The Genetic Algorithm

The proposed system is divided into several modules, and it operates primarily a classification in different families of the infrasound explosive events. Subsequently, a module called *features exstractor*, gets from each event special parameters, starting from the waveform (features). A third and final module takes care to find, in a multitude of possible features extracted, those that best characterize the classification above. For this step we use the theory of genetic algorithms (GA), as features selectors. In general to apply GA modules to a specific problem, it is necessary to encode properly a general solution of the problem into a chromosome and then define an appropriate *objective function*. This is achievable if it is possible to make a good set of both of these two characteristics; each of them is considered as a binary gene and each individual is represented by a binary string, or mask, where the length is determined by the number of extracted features.

	F1	...	F_k	...	F8
M	...	...	m_k	...	...

Fig. 7 Bit Mask of the used GA. The value m_k is 1 if the feature is valid, 0 otherwise

Every bit inside the mask will be 1 in value, if the corresponding feature is valid, zero otherwise. Another crucial step is the definition of the *object function* that is defined in order to represent the distance between an event belonging to a class and the corresponding centroid of that class. In this special case, we work with five different classes, consisting of several identified events.

C. Results

Distributed GA has been implemented in C language by using the PGAPack.
Inside the package it possible to use GA on a single core, but also to use them in parallel for single or multiple processors. In this case, we used Message Passing Interface (MPI) library. In particular we use the MPICH2 module inside PI2S2 grid network with good results. Every job of the grid runs a number of N_p tests, and with massive use of MPI, each node processes a single class.
The graphs below illustrate the obtained results on the analyzed data for each class. Even if the number of simulations is the same, the calculation time is different for different instances, because the number of elements that make up the each class is different.

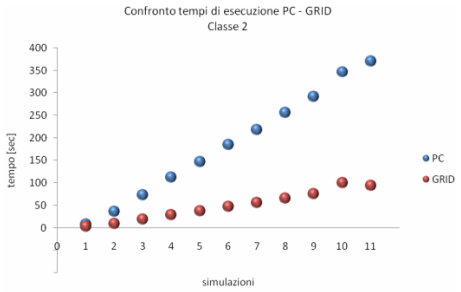


Fig. 8 Execution time (Class 2) on a single PC vs Grid network for different simulations.

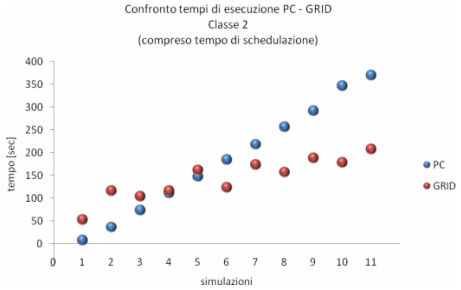


Fig. 8 Graph of the total time (Class 2)on a single PC vs Grid network for different simulations. In some cases few advantages are visible.

Figures show that execution time of the genetic algorithm on the GRID is considerably smaller than the time requested on a single PC. This reduction becomes more evident using heavier simulations, formed by a great number of tests. So, by optimizing parallel code and by using the Grid, it is possible to obtain a significant improvement in execution time of the algorithm. Even if scheduling time could affect Grid performance for small series, the GA reaches its best performance if the number of simulations increase.

IV. Conclusion

In this work the performance of different algorithms have been evaluated both on a single PC and on a local cluster. Although we refer to two analysis software that have very different specificities, it is possible to identify general elements when we approached the world of the advanced computation. First of all it is useful to remember that the software could be developed in native parallel mode or made parallel, by using some artifices. In fact it is possible to take advantages using job farming method if the number of nodes and executions increase also for sequential algorithms. In all discussed cases we consider the delay time introduced by the scheduling time of each job. So, the average scheduling time can be considered an offset added to the execution time itself. In some cases the total delay is almost compensated by the increased speed of execution of the job within the Grid network. Where special activity like monitoring and surveillance of active volcanoes is required, it is good to use special algorithms like the PI2S2 *Emergency Queue.*

Acknowledgments

The authors would like to thank to A. Drago, F. Pistagna, S. Lizzio, M. La Via and

S. Mangiagli for their contribution.

REFERENCES

[1] A.Cannata, P. Montalto, E. Privitera, G. Russo, S. Gresta. "Tracking eruptive phenomena by infrasound: May 13, 2008 eruption at Mt. Etna", submitted to GRL

[2] A. Cannata, P. Montalto, E. Privitera, G. Russo."Characterization and location of infrasonic sources in active volcanoes: Mt. Etna, September-November 2007", submitted to JGR

[3] Andrea Cannata, Placido Montalto, Eugenio Privitera, Danilo Reitano, Gaetano Russo."Classificazione di eventi infrasonici sul vulcano Etna" Oral presentation, Grid Open Day alla sezione di Catania dell'INGV, 2008.

[4] A.Cannata, P. Montalto, E. Privitera, G. Russo, S. Gresta. "Tracking eruptive phenomena by infrasound: May 13, 2008 eruption at Mt. Etna", Sezione Poster, AGU 2009

[5] A. Cannata, P. Montalto, E. Privitera, G. Russo."Characterization and location of infrasonic sources in active volcanoes: Mt. Etna, September-November 2007",Sezione Poster, AGU 2009

[6] M. Soryani, N. Rafat: "Application of genetic algorithms to feature subset selection in a farsi OCR", Proceedings of word academy of science, 2006.

[7] H. Vafaie, K. De Jong: "Genetic Algorithms as a tool for features selection in machine learning", George Mason University, 1992.

[8] J. Jarmulak, S. Craw: "Genetic algorithms for features selection and weighting", The Robert Gordon University, 1999.

[9] A. Jain, D. Zongker: "Feature Section: Evaluation, Application, and small sample performance", Michigan State University, 1995.

[10] M. Jamshidi: " Tools for intelligent control: fuzzy controllers, neural network and genetic algorithms", The University of New Mexico, 2003.

[11] R. C. Gonzalez, R.E. Woods, Digital Image Processing 2nd Edition, Prentice Hall, New Jersey, 2002.

[12] R.O. Duda, P.E. Hart, D.G. Stork, Pattern classification 2ed., Wiley, 2000.

[13] I. Foster, C. Kesselamn, S. Tuecke: " The anatomy of the Grid; Enabling scalable virtual organizations", Intil J. Supercomputer Applications, 2001.

[14] I Foster: " What is the Grid? A three point checklist", Talk from I.Foster homepage, 2002.

PON CyberSar Project Results on Drought Mitigation in Complex Water Systems

G.M. Sechi[1] and A. Sulis[1]
[1]*Department of Land Engineering, University of Cagliari (Italy)*
{sechi,asulis}@unica.it

***Abstract* — This paper presents the methodology developed and the results obtained at the University of Cagliari (Italy) under the PON-CyberSar Project (National Operation Programme Scientific Research, Technological Development, Higher Training), Activity 11.2 (High Performance in Data Process and Simulation) in the field of planning and management of complex water resource systems. Results demonstrate the significant benefit of combining traditional tools for system analysis and Grid Computing. Remarks on hindrances of the Grid approach are also highlighted in the conclusions.**

***Index Te1ms*— CyberSar, Complex Water Systems, Simulation Analysis, WARGI-DSS.**

I. INTRODUCTION

PARTICULARLY in water systems that frequently experience severe drought events, the definition of planning and management plans must be proactive. Simulation and optimization models within Decision Support Systems (DSSs) provide the main tool for researchers and practitioners to analyze the behavior of these systems. In the case of computationally intensive software implementation, simulation and optimization models can result extremely time and resources consuming and the applicability of these models is sometimes limited to relatively simple applications. Grid provides the ability to solve problems too big for any single supercomputer with the flexibility to work on multiple smaller problems in front of a home computer. This is why Grid computing is attractive to DSSs for definition and evaluation of drought mitigation plans. WARGI (WAter Resource Graphical Interface) is a generic DSS for planning and management complex water systems developed at the University of Cagliari (Italy), specifically to meet the growing demands in multireservoir system under water scarcity conditions. This paper focuses on the applicability to complex real systems of the combined simulation and optimization approach in WARGI with the Grid technology for the definition of drought mitigation measures and reservoir operating rules based on a large number of possible future system evolutions. The primary scope of this article is to summarize the objectives, results and products obtained under the PON-CyberSar by the Unit of the University of Cagliari working in the Activity 11.2. In-depth explanations of the practical application of the proposed techniques and their impacts on system management are beyond this primary scope. Dissemination of the objectives, results and products to a larger scientific community than that specifically involved in this project, has been done through publications in international journals ([11], [12]) and in proceedings of conferences ([9], [13]).

Briefly, two main approaches of Grid computing in water resource system have been developed by the Authors. Both approaches focus on the combined use of the linear programming model and the simulation model within WARGI-DSS ([8]). One approach deals with the implementation of the implicit stochastic optimization for the identification and the evaluation of linear operating rules in

multireservoir systems. The approach cycles through the optimization, regression and simulation models to refine the operating rules. The second approach was proposed with the aim of identifying and evaluating mitigation measures in a proactive approach that anticipates the trigger of these measures. Operating rules and mitigation measures are based on the value of the state parameters of the water system being analyzed, among which the current or expected hydrological values.

Particularly in complex systems, the hydrology is affected by a high level of uncertainty, and operating rules and mitigation measures must be defined based on a large set of hydrological series (short observed series or generated series) the system may experience. In such cases, the application of the combined optimization-simulation technique may be – time and –resource consuming. The divisibility of the problem gives us the opportunity to solve computational problems using unused resources/services provided by Grids.

The rest of the paper is organized as follow. Section II reviews the state of the art of integrated optimization and simulation techniques particularly related to multireservoir operations. In Section 3, an overview of the simulation and linear optimization models within the WARGI-DSS is presented. In what follows, some aspect of the Grid computing environment and the porting of the proposed approaches are described fist (Section 4). The approaches are then applied to a complex water system in Southern Sardinia (Italy) for the definition and evaluation of reservoir operating rules and drought mitigation measures in a proactive approach (Section 5). Finally, some remarks on hindrances of the Grid approach are also highlighted in the Section 5. Appendix A introduces some details of the combined optimization-simulation approach for drought measures.

II. OPTIMIZATION AND SIMULATION FOR MULTIRESERVOIR SYSTEMS

Reservoirs regulate surface flow for allocation of water resources to meet the spatial and temporal demand for multiple uses. A decision-making procedure is needed for reservoir operation to balance demand and supply for optimal economic and social benefits. Operating rules are used to guide operators when it is not possible to satisfy ideal storage levels and downstream releases. Ideal storage volumes in individual or multiple reservoirs are typically defined by rule curves. When conditions are not ideal, operating rules define what should be done for various combinations of system states and hydrological conditions. The purpose of operating rules is to distribute any necessary deviations from ideal conditions in a way that minimizes the total perceived discomfort to all water users in the system.

The operating rules can be found from optimization and simulation methods and various models based on these methods have been proposed and reviewed by many authors ([15], [10], [14], [2]). In general terms, simulation methods for the analysis of water systems behaviors is often the only method for dealing with large and complex problems that cannot be reproduced by experiment on actual systems or by analytical solutions. Simulation is a widely accepted technique to aid in both the design and operation of water resource systems. Generally simulation models provide the response of a system with an assigned set of resource characteristics, resource availabilities and demands and predefined operating rules. Simulation models work well only when a relatively small number of alternative combinations of physical facilities and operating procedures need to be evaluated. Unfortunately, in complex water resource systems the number is quite large and the 'trial and error' process of simulation becomes very time consuming. The process of employing optimization to reduce the range of designs and policies requiring simulation and more in-depth evaluation is often called "preliminary screening" ([4]). In most studies, a three-phase

approach has been used for development of reservoir operating rules. This approach cycles through an optimization model, a multiple regression analysis, and a simulation model. The optimization model determines a set of reservoir releases maximizing or minimizing an objective function subject to a set of constraints. Optimal releases are then regressed to determine the monthly operating rules as a function of actual or forecasted inflows to the reservoirs and the amount of water storage in the reservoirs. The simulation model refines and evaluates the operating rules.

The choice of the optimization model depends on characteristics of the system being considered, on availability of the data, and on the objectives and constraints specified. The operating rules can be found from stochastic or deterministic models. One issue with stochastic optimization models is the assumption of full distribution knowledge in each and every of uncertain data. As such information may rarely be available in practice some works have focused on the use of alternative optimization approaches where the uncertainties addressed are non-stochastic. An approach is to employ deterministic optimization models to what is a probabilistic problem. This is a method of implicit stochastic optimization, where most stochastic aspects of the problem are implicitly included and operating rules are found from deterministic optimization using linear or nonlinear regression procedure.

Previous review of reservoir operating rules include [5] and [3]. A wide variety of parameterized operating rules are available among which the linear decision rule (LDR) is the simplest. LDR is a rule with a linear functional form and having a decision parameter on which the rule is based. LDR may be expressed as:

$$R_t = S_t - b$$

Where R_t is the release in the period t of reservoir operation, S_t is the storage of reservoir at that period, b is the deterministic parameter chosen to optimize the objective function of the optimization model. The optimization model with the use of LDR was first introduced by [6] and is of concern in the proposed approach.

III. OPTIMIZATION AND SIMULATION IN WARGI-DSS

WARGI (WAter Resources Graphical Interface) is a generic DSS for planning and management complex water systems developed at the University of Cagliari (Italy). The DSS is specifically developed to meet the system management requirements to satisfy the growing demands in multi-reservoir systems under water scarcity conditions, as frequently, happen in the Mediterranean regions. The WARGI modeling capability includes several interrelated macro-modules implemented in C++ and Tcl/Tk (Figure 1); the main ones being: the graphical user interface module, the system initialization and input module, the simulation module (WARGI-SIM), the deterministic optimization module (WARGI-OPT), the quality optimization module (WARGI-QUAL), the scenario optimization module (WARGI-SCEN) and the output plotting module.

In WARGI-DSS, the physical system in a single period ∂ is represented by a direct network, the basic graph. WARGI-OPT generates a dynamic multi-period network replicating the basic graph for each time period ∂=1,…,Δ and then connecting the corresponding reservoir nodes for consecutive time periods. The time period ∂ can be hourly, daily, monthly or even seasonal, depending on the specific application. In the network representation, water deficits are represented by additional flows from dummy nodes (that is external system resources). WARGI-OPT is a linear programming model that minimize the totals costs in the system including operative, maintenance and replace costs (OMR), costs of new works and deficit costs based on demand priorities. Constraints in the model include the continuity equations at the nodes and relations between flow and project variables.

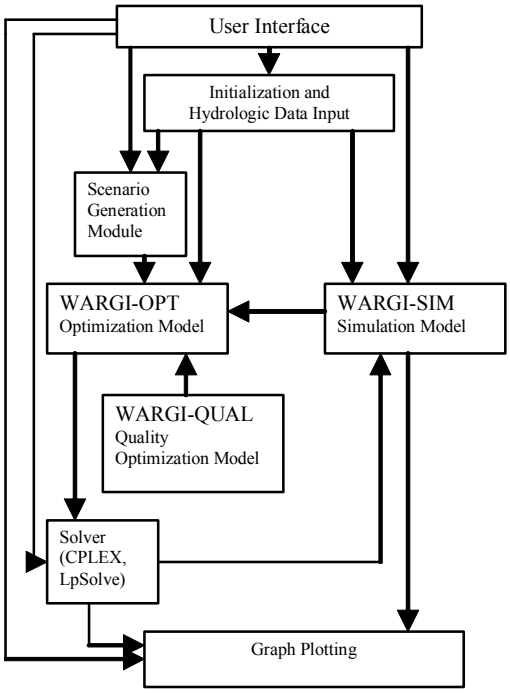


Figure 1. The structure of WARGI-DSS (adapted from [7]).

In WARGI-SIM, the operator can define preferences for each combination of possible transfers between resource and demand node. These preferences may be based on physical, economic, legal, contractual, quality considerations or operator judgment. Each demand has a hierarchical list of resources from which a supply flow could be activated to meet the demand. Each reservoir and aquifer has a reserved zone and a target storage volume. The operator divides the volume required by each demand into priority zones. Following priority ranking, the supply procedure defines flows in the network from resources to demands. [11] has proposed the implementation of an ISO approach in WARGI-DSS and the porting of that approach to a Grid environment. Optimization, regression and simulation models were used to identify and evaluate linear operating rules in complex water system based on large sets of hydrological series.

[8] have recently developed a full integration of WARGI-SIM and WARGI-OPT in the DSS. This mixed simulation-optimization approach was proposed with the aim of identifying and evaluating mitigation measures in a proactive approach that anticipates the trigger of these actions.

IV. GRIDS IN WATER SYSTEM ANALYSIS

A common disadvantage of the traditional optimization or simulation modelling approaches is the large number of iterations required to achieve acceptable levels of confidence regarding model data uncertainties. Particularly in the case of complex water systems with many resources serving many demands, these models can result extremely time and resources consuming and their applicability can be sometimes limited to relatively simple applications. Today, Grid computing uses the resources of a large number of separate computers connected by a network to solve large-scale computation problems. Grid provides the ability to solve problems too big for any single supercomputer with the flexibility to work on multiple smaller problems in front of a home computer. This is why Grid computing is attractive to geographically-distributes collaborative researches in the environmental sciences.

[11] presented the details of the porting of the linear optimization model WARGI-OPT and the simulation model WARGI-SIM. Here some of these aspects are briefly recalled. In WARGI, the linear optimization model associated with each hydrological series determines the construction of a file in MPS (Mathematical Programming System) format representing a single job. WARGI-OPT organizes the set of independent jobs in a job collection with a root node and leaf nodes. The collection execution on the Grid is described by a jdl file with desired collection characteristics and constraints, which are used by the matchmaking process to select the resources that best satisfy the collection requirements. The free linear programming

solver LpSolve (http://lpsolve.sourceforge.net/) is not invoked directly with the Executable attribute, but runs within a script (indicated in the Arguments), which correctly sets its environment. As LpSolve is less than 10 MB (standard dimension of the Input Sandbox), it is carried out within the Input Sandbox that also contains a list of mps files that the user wants to be downloaded on the Computing Element. The Output Sandbox contains output and error files uploaded through the Grid FTP service. Output files include the flow configurations provided by LpSolve.

This architecture originally proposed by [11] has been implemented in two different approaches to define multireservoir linear operating rules and drought mitigation measures in a proactive approach. In both approaches the decision variables to be evaluated are based on a large number of hydrological series the system may experience. In the analysis of the water system over a time horizon of T periods, a large set of hydrological series are generated to account for the high uncertainty in hydrological parameters. At each instant τ_i in the simulation process, WARGI-OPT created a constant number of optimization problems equal to a selected sub-set of generated hydrological series. Specifically in the Grid environment, the optimization problems were organized in a job collection with a constant number of leaf nodes equal to the selected hydrological sub-set.

In the approach proposed by [11], WARGI-OPT defines drought mitigation measures to anticipate the occurrence of water scarcity. These pre-emptive measures are developed in a proactive approach to face drought events ([16], [7]). The pre-emptive actions are based on results provided by WARGI-OPT considering synthetic hydrological scenarios. Flow configuration provided by WARGI-OPT can be seen as a reference target for the simulation phase and WARGI-SIM can operate to minimize overall system management costs. Figure 2 schematizes this interaction between WARGI-OPT and WARGI-SIM in the analysis of the water system on a time horizon of T periods. At each instant τ_i in the simulation process, WARGI-OPT forecasts system evolution on a time horizon Δ, shorter than simulation time horizon T. This forecast is based on a hydrological scenario (water resource input) and a demand scenario (water resource output). Based on information provided by WARGI-OPT at each instant τ_i, WARGI-SIM defines a set of pre-emptive measures which modify system management in the subsequent periods until τ_{i+1}. Thus, simulation explores the system's behavior dynamically using a set of new measures identified by optimization to handle resource shortage events. Details of the mathematical model are presented in the Appendix A.

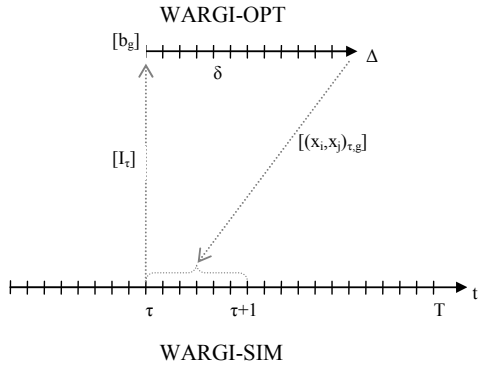


Figure 2. Mixed optimization-simulation approach for drought mitigation measures in WARGI-DSS.

In Sulis (2009), a multiple regression model is applied to the optimum storage trajectories obtained by WARGI-OPT. A lag-1 linear regression can be considered suitable for the analysis of many water systems. The promising regression based operating rules is in a linear form (LDR). [1] concluded that simple correlations of releases with selected independent variables as storage volumes are useful in a preliminary selection of the form of linear operating rules. Also, the coefficients in the regression equations may assist in the understanding of the physical behavior of the

system under consideration. The LDR is then tested in WARGI-SIM. The approach cycles through WARGI-OPT, the regression model and WARGI-SIM until further iteration do not further improve the value of system performances (Figure 3).

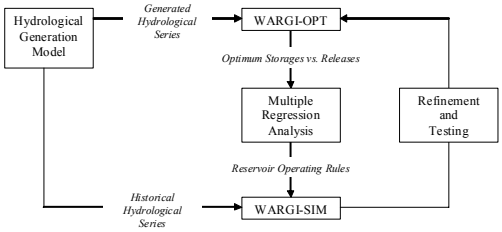


Figure 3. ISO approach for operating rules in WARGI-DSS (adapted from [11]).

The appropriateness of operating rules should be estimated using the values of reliability, resilience and vulnerability of the system. [1] found that the correlation coefficient (R^2) does not always produce the best operational model as measured by simulation results. Thus simple operating rules with a lower R^{22} may be more appropriate than policies with higher R^2.

V. CASE STUDIES

The proposed approaches in the Grid environment have been applied to a water system located in Southern Sardinia. Results in term of time saving have been compared with the same methodology working in a local environment. The Flumendosa-Campidano water system (Figure 4) is located in the South Sardinia, Italy. The hydrologic regime is characterized by significant variation in both monthly and annual flows. The system is composed by 10 reservoirs that are expected to regulate the unstable streamflows and to assure sustainable supplies to nine urban, two industrial and eleven irrigation demands.

The increase in demand (particularly urban and agriculture) and the intense water scarcity events verified in recent years, require the water authority to revive the management plan with new reservoir operating rules and drought mitigation plans to guarantee fulfillment of high priority demands during water scarcity conditions. The success of these plans hinges on increasing the water system reliability, which is subject to hydrologic uncertainty probably related with long-term climate changes.

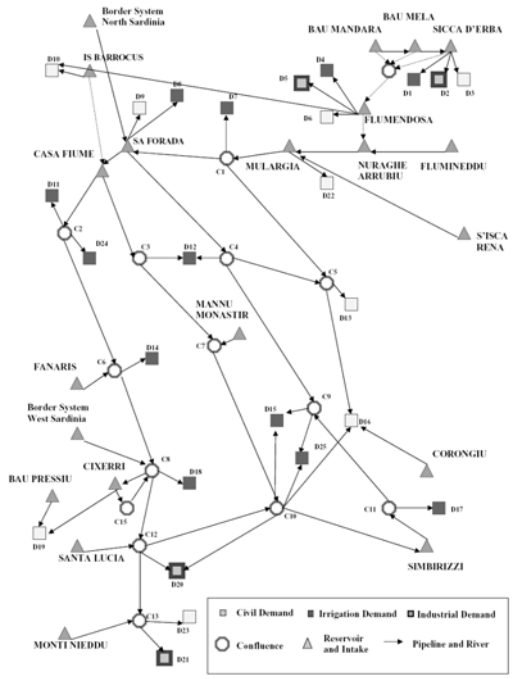


Figure 4. Network representation of the Flumendosa-Campidano water system.

In [11], a lag-1 linear regression was considered suitable for the analysis of the Flumendosa-Campidano water system. Based on the observed 54-year hydrological series, 500 synthetic hydrological series 1 year in length with different probabilities were generated.

Table 1 shows proactive and reactive measures currently applied and planned in the Flumendosa-Campidano water system. In [12], the WARGI DSS has been applied to the system to provide the water authority of the Flumendosa-Campidano with a first estimate of the relationship between hydrologic uncertain

data and drought mitigation measures. In the DSS application presented by [7], the proactive measures were selected based on the information provided by WARGI-OPT under seven hydrological scenarios 2 year length with different criticalities in term of annual mean value.

Table 1. General classification of drought mitigation measures currently applied or planned.

Measure	Proactive	Reactive
Supply Increase	Increase of available resources	Use of additional sources
Demand Reduction	Pricing and Use Reallocation Plans,	Restrictions and Rationing
Impact Minimization	Warning Systems and Education Activities	Temporary Resource Reallocation

In the local approach, WARGI-DSS ran on an Intel Pentium with Single CPU clocked at 1.60GHz having 1GB RAM. Results on computational time of the combined optimization approach for drought mitigation measures and of the ISO approach for reservoir operating rules have been presented in [12] and [11], respectively. Here results from [12] are briefly summarized in what follows. Figure 5 shows the trend of computational time of the combined optimization-simulation approach when up to seven hydrological series are considered. The computation time in the local approach vastly increased and the software simulated the system in about 21min when seven hydrological series were considered in the optimization phase. At that time, there were 10 Intel Pentium Dual CPU running Scientific Linux SLC in the Grid authorized for a given job. The total Grid computation time is defined by the Grid latency (the time needed to submit a complete set of jobs, and collect a complete set of results), the waiting time in case a job is delayed in the Grid queues and the running time on the nodes. Figure 5 shows that the total Grid computation time at different hydrological series remains approximately stable (~ 8 min). The Grid approach seems promising for more than 2 hydrological series and the benefit improved as the number of hydrological series increases. The total computation time was reduced more than 60% in the case of seven hydrological series.

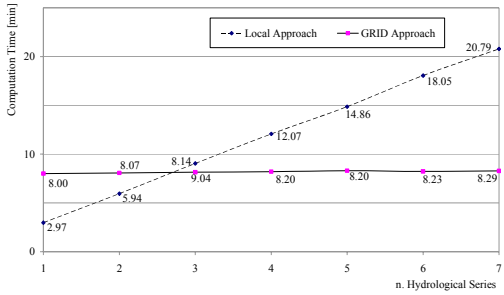


Figure 5. Computation times for 54 years simulation in local and Grid approaches (adapted from [12]).

Results demonstrate that Grid computing is a promising approach for scaled-up simulation analysis of complex water system with high uncertainty on hydrological series. However in the authors' view, there are still some hindrances for an extensive use in the water system field. Most importantly, the availability of a friendly graphical user interface between users and Middleware could provide the access to Grid resources and services to a larger non computer expertise community.

APPENDIX A

In the analysis of a water system for a time horizon T with a time step t, WARGI-OPT forecasts the system evolution on a time horizon Δ at each synchronization period τ_i based on the current water storages in the system's reservoirs and a user-selected future hydrological synthetic scenario g. The linear optimization model is in the following form:

$$\min_{t=(\tau,\ \tau+\Delta)} \quad [c_\gamma Y + c_i x_i + c_j x_j] \qquad (1)$$

$$\text{s.t.} \quad A[x_i, x_j] = b_g \qquad (2)$$

$$F\left(Y,[x_i,x_j]\right)\geq 0 \qquad (3)$$

$$l\leq[x_i,x_j]\leq u \qquad (4)$$

The set of costs $[c_\gamma]$ is related to the project variables [Y], c_i represents operative, maintenance, and replace costs (OMR) or user-defined costs along transfer arcs, and c_j represents deficit costs based on demand priority ranking. The variables $[x_i, x_j]$ are the subsets of the flow variables x related to flows along the multi-period network and to flows along the deficit arcs, respectively. The parameters l and u are the lower and upper bounds on $[x_i, x_j]$.

WARGI-SIM uses the variables $[x_i, x_j]$ provided by WARGI-OPT at each synchronization period τ_i and the preferences and priorities [v] provided by the users to set up the proactive mitigation measures $[z_\tau]$:

$$z_\tau = f_1\left(\left[x_i,x_j\right],\nu\right)_\tau \qquad \tau=\tau_1,\tau_n \qquad (5)$$

These preemptive measures $[z_\tau]$ can modify the water allocations (X_t) from those previously defined using the allocation rules [r] and user-defined preferences and priorities [v].

Consequently, during the subsequent periods until τ_{i+1}:

$$X_t = f_2\left(z_\tau,\nu,r\right) \qquad t=(\tau_i,\tau_{i+1}) \qquad (6)$$

ACKNOWLEDGMENTS

The authors would like to thank the Istituto Nazionale di Fisica Nucleare (INFN) di Catania for its support in the WARGI-DSS porting to the Italian National Grid.

REFERENCES

[1] Bhaskar, N.R., Whitlatch Jr., E.E., "Derivation of Monthly Reservoir Release Policies", Water Resources Research 16(6), 987-993, 1980.

[2] Labadie, J.W., "Optimal Operation of Multireservoir Systems: State-of-the-Art Review", Journal of Water Resources Planning and Management 130(2), 93-111, 2004.

[3] Loucks, D. P., Sigvaldason, O.T., "Multiple-reservoirs Operation in North America", In: The Operation of Multiple Reservoir Systems - IIASA Collaborative Proceedings Series CP-82-S3 (ed. Kaczmarek, Z., Kindler, J.), pp. 1-104, Institute for Applied Systems Analysis, Laxenburg, Austria, 1982.

[4] Loucks, D.P., van Beek, E., Water Resources Systems Planning and Management: An Introduction to Methods, Models and Applications, UNESCO Press, Paris, 2005.

[5] Lund, J.R., Guzman, J., "Derived operating rules for reservoir in series or in parallel", Journal of Water Resources Planning and Management 125(3), 143-153, 1999.

[6] Revelle, C., "The Linear Decision Rule in Reservoir Management and Design, I Development of the Stochastic Model", Water Resources Research 5(4), 767-777, 1969.

[7] Rossi, G., "Drought mitigation measures: a comprehensive framework", In: Drought and drought mitigation in Europe (ed. Vogt, J.V., Somma, F.), pp. 233–246, Kluwer, The Netherlands, 2000.

[8] Sechi, G.M., Sulis, A., "Mixed simulation-optimization technique for complex water resources systems analysis under drought conditions", Chapter 11, In: Methods and Tools for Drought Analysis and Management (ed. Rossi, G., Vega, T., Bonaccorso, B.), pp. 217-237, Springer, Berlin, 2007.

[9] Sechi, G.M., Sulis, A., "Un approccio Grid per la mitigazione della siccità in sistema idrici complessi", Conferenza Nazionale "Italian E-Science 2008", Napoli, 2008.

[10] Simonovic, S.P., "Reservoir Systems Analysis: Closing Gap between Theory and Practice", Journal of Water Resources Planning and Management 118(3), 262-280, 1992.

[11] Sulis, A., "Grid computing approach for multireservoir operating rules with uncertainty", Environmental Modelling & Software, doi:10.1016/j.envsoft.2008.11.003, 2009.

[12] Sulis, A., Ardizzone, V., Giorgio, E., Sechi, G.M., "The porting of WARGI-DSS to Grid computing environment for drought plans in complex water systems", Journal of HydroInformatics, submitted.

[13] Sulis, A., Ardizzone, V., Giorgio, E., Sechi, G.M., "The porting of WARGI-DSS to Grid environment for planning and management in complex water systems", 33rd IAHR 2009 Congress: Water Engineering for a Sustainable Environment, Vancouver, Canada, 2009 (accepted).

[14] Wurbs, R.A., "Reservoir-System Simulation and Optimization Models", Journal of Water Resources Planning and Management 119(4), 455-472, 1993.

[15] Yeh, W.W.-G., "Reservoir Management and Operation Models: A State-of-the-Art-Review", Water Resources Research 21(12), 1797-1818, 1985.

[16] Yevjevich, V., Hall, W., Salas, J., Coping with Drought. Water Resources Publication, Littleton, Colorado, 1983.

VisualGrid: Streaming and Visualization for Grid Applications Monitoring in gLite based Infrastructures

G. Minutoli[1], M. Paone[2], M. Fazio[1], G. Iellamo[2] and A. Puliafito[2]
[1]Engineering Faculty, University of Messina, Italy
[2]Consorzio COMETA, Catania, Italy
{gminutoli, mpaone, mfazio, giellamo, apuliafito}@unime.it

***Abstract*— The facility of monitoring users' applications represents an important task for effective functionality of a Grid infrastructure. In this paper, we describe the development of *VisualGrid*: an application which - exploiting the common techniques of scientific visualization and streaming - allows users of a gLite based Grid infrastructure to monitor the evolution of their applications. Technical and practical problems encountered during the development are analysed, and a performance evaluation of the developed prototype is presented.**

***Index Terms*—Grid, Monitoring, Scientific Visualization, Streaming, gLite**

I. INTRODUCTION

Nowadays, Grid Computing based systems are often used to execute applications mainly related to engineering, Earth science, bio-informatics, chemistry, physics and astrophysics. Fluid-dynamic algorithms, simulations of aerodynamics problems, software performance analysis tools and thermodynamic or electromagnetic simulations represent some examples of the most popular applications executed on Grid, making optimal use of parallel processing capabilities, with huge benefits in terms of time and performance.

The facility of monitoring users' applications represents an important requirement in order to guarantee the effective functionality of a Grid infrastructure. The most part of the common Grid middleware accomplishes it adopting different software solutions. Nevertheless, the implemented monitoring facilities provide users by the system status of their applications. They can only determine if their applications are scheduled, running, finished or aborted. However, whilst this kind of information is completely enough from the point of view of Grid management, it is not enough for the users that often need a more specific information. Let us consider the typical example of a Monte Carlo event simulation for LHC experiments. Usually, a single application generates thousand of events for 5-10 hours of CPU. In order to control this generation process, a user has to check, from time to time, the current number of generated events. However, there are no ways to provide the user with such an information in Grid [1].

In this paper, we propose the use of *VisualGrid* to enable users of a gLite middleware based Grid infrastructure to monitor the evolution of their applications in a simply and intuitive manner by using the common techniques of scientific visualization and streaming. Starting from the key idea a picture can speak a thousand words where tables of numbers can say very little, we consider that once a user submits his application on the Grid, an image representing the application run-time status is generated (at regular intervals) and recorded on storage elements of the Grid system. Whenever the user wants to monitor the workflow of his

application, all the images produced up to that point are recovered by one or more Grid nodes and joined into a video that will be delivered to the user in streaming on the network.

The use of a solution such as *VisualGrid* may be very useful, especially in presence of applications characterized by a long term execution: users can save time by not waiting for applications that do not work as they should. Moreover, since the *VisualGrid* architecture is totally job-oriented, it does not require neither software pre-installation nor any modification of the adopted Grid middleware, allowing also to indistinctly exploit all the storage and computational resources offered by the overall Grid system with huge benefits in terms of scalability.

In the rest of this paper, the implementation of *VisualGrid* on a Grid system with gLite middleware is depicted providing also some performance measures able to demonstrate its feasibility. In particular, in section II we review some basic concepts about the Grid middleware adopted. The section III discusses broadly the architecture of the proposed system highlighting its objectives and functionalities. In section IV technical details about the implementation of *VisualGrid* are described, while experimental assessments are presented in section V. Finally, concluding remarks are given in section VI.

II. gLite Middleware

Based on Internet the "World Wide Web", or simply the "Web" as ordinary it is called, lets people to share the universe of network-accessible information distributed on millions of computers around the world. Similarly Grid Computing allows the shared and coordinated use of heterogeneous resources in dynamic virtual organizations: namely set of individuals and/or institutions which, defined by sharing rules, do not exchange files but share direct access to computers, software, data, and other resources geographically distributed and belonging to distinct administrative domains [2].

The core of this technology is a software layer, called *Grid Middleware*, that placed between individual resources and applications provides the latter with the abstraction of a single computing platform hiding the heterogeneity, the distribution, and ownership of the involved resources. Modules for authentication, authorization, monitoring, Grid resource discovery, processes scheduling and dispatching are some of the several components which it provides [3].

gLite represents one of the most common middleware used in Grid environments [4]. Developed in the context EGEE project [5], it is composed of a set of integrated components, some developed inside of EGEE, others coming from various related projects, such as Condor [6], Globus [7], LCG [8], Virtual Data Toolkit (VDT) [9].

An infrastructure based on this middleware (see Fig. 1) consists of two kinds of sites: *Resource Sites*, which provide computational and storage resources, and *Central Operation Sites* which host the central services needed for components interconnection between *Resource Sites* [3]. In general the following set of services result distributed on the different sites:

- **Basic Services**.
  These includes many of the elements of a gLite infrastructure that are in charge of handling jobs as well storage. The computational resources localized at each *Resource Site* are referred to as Computing Elements. Each of them includes both one or more Worker Nodes (WNs), where

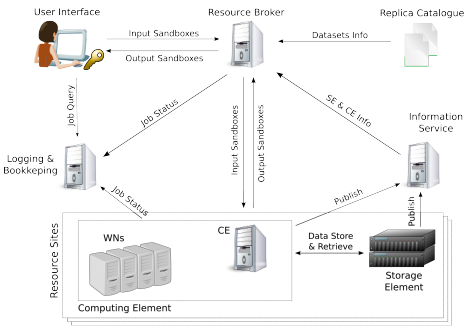


Fig. 1. Main components of the gLite architecture.

computation is carried out, and a machine (CE) which provides an interface to the site resources (Grid Gate) and a Local Resource Management System (LRMS). In general the Grid Gate accepts jobs that are dispatched by the LRMS for the execution on different WNs. Each WN can execute in parallel a number of jobs equal to the number of Grid slots it has. This number can be configured, but usually it is set equal to the number of available CPUs. The term Storage Elements (SEs) is used, instead, to identify the storage resources. They are designed to provide a generic interface to storage resources such as disk servers, arrays or tape systems.

- **Global Services**.
  These are services used across the entire Grid infrastructure. The most important are:
  - The *Grid Security Infrastructure* that takes care of authentication and authorization: it is based on the X.509 public key infrastructure and on the Secure Socket Layer (SSL) and supports single sign on and delegation.
  - The *Information Service* (IS) that provides information about the status of the Grid shared resources.
  - The *Workload Management System* that performs the process of matching the job to the most appropriate resource taken from the overall Grid resource pool, to which the user has access permission (*matchmaking*). This service runs on a machine called Resource Broker (RB).
  - The *Logging and Bookkeeping* (LB) service that logs all the information about the jobs managed by the workload management system.
  - The *Replica Catalog* that contains a map of the data scattered on the Grid. Such service is a very important element because, in the Grid, files could have more than one replica and users have to access them without knowing their exact location. The catalog used in gLite is called LCG File Catalog (LFC) and it is accessible using the data management tools.

The key concept on which gLite is based is the *"Job Submission"*. A job is an entity which contains all the stuff needed for a remote execution such as the executable program, the environment settings and the Input/Output data. All these parameters are exposed in a text file written in the *Job Description Language* (*JDL*), that is a high-level language based on the *Classified Advertisement* formalism [10].

In particular, when a user wants to execute a job on the Grid he can access the infrastructure through systems configured with the middleware environment and called User Interfaces (UIs). Once the JDL document is created, the job is submitted to the RB: all the files needed for the remote execution are packed in the Input Sandbox. The RB analyses the JDL file and selects the most appropriate resources in respect to the job requirements and to the current Grid status. After that, the Input Sandbox is sent to the selected CE for the execution. Once the task has been accomplished, all the output files are packed in the Output Sandbox and sent back to the RB. Finally, the user can retrieve the Output Sandbox onto his UI.

III. Architecture

Based on the use of computer graphic methods and techniques to create visual images, the scientific visualization represents a valuable instrument which helps researchers in understanding complex, often massive numerical representation of scientific results. Static two-dimensional, three-dimensional plots, volume rendering, colour transformations, isolines and isosurfaces, represent some of the most popular techniques used to analyse and display the ever growing amount of experimental data collected.

The key idea of *VisualGrid* is to use the scientific visualization in order to allow users to track the run-time status of their applications running on the Grid. The description of

the whole application, to better illustrate the developed architecture, can be restricted to two main activities referred, respectively, as the *"Images Generation"* and *"Images Streaming"* stages. In the following, such activities are

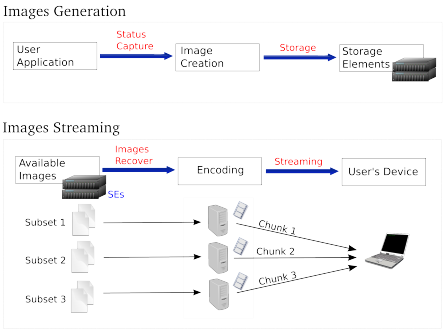


Fig. 2. *VisualGrid* activities with their corresponding phases

explained highlighting the phases they consist of, as shown in Fig. 2.

A. Images Generation

The *Images Generation* activity deals with the creation and the storage of images housing the information about the evolution of the user applications. First of all, a user submits his application on the Grid. At regular intervals an image representing the application internal status is generated, sent to the Grid storage system and registered on the file catalog.

B. Images Streaming

The *Images Streaming* activity, instead, deals with the end user requests and the applications monitoring through streaming. Whenever the user wants to monitor the workflow of his application, he submits his request using a GUI, and specifying the application to be monitored and the characteristics (in terms of resolution, frame rate, colour depth, video format) that the multimedia content generated for the streaming must have. All the images produced up to that point are recovered and processed by one or more Grid nodes, according to the requested characteristics by the user. The use of more jobs allows to reduce the time interval between the submission of monitoring request and the start up of the streaming phase. In this vision, each worker node will process only a subset of all the available images, creating a single chunk of the entire multimedia content. Once all chunks have been created, the streaming phase allows to the user to watch the entire stream in a transparent way over a set of plain HTTP connections. The media player is not aware of the whole *VisualGrid* architecture, in fact the application automatically composes a playlist that contains the proper association of each Grid node to the correct video chunk. The HTTP protocol has been chosen because, although the highly distributed nature of *VisualGrid*, the connections based on this protocol guarantee the best behaviour with firewalls and network address translations (NATs).

IV. The *VisualGrid* gLite Implementation

In this section we describe the gLite implementation of *VisualGrid* analysing the different issues encountered and the way we tackled them. More precisely, different technical details of the application design are discussed covering the aspects related to both the *Images Generation* and *Images Streaming* activities.

First of all, it is important to distinguish between programs running on the UI and programs running on the WNs. The former are standard standalone software applications that interact with the API provided by the middleware, while the latter are software applications encapsulated in Grid jobs.

The *Images Generation* activity is carried out by the *VisualGrid Producer* module. Integrated as a single script into the user application, it is executed as a separate process on the same worker node (called for convenience *Application Worker Node*). This module is responsible to create the images on the application status and to store them on the Grid SEs. A large number of tools - some open source, others commercial - can be exploited to accomplish

this task. Gnuplot, Scilab, MayaVi and OpenDX represent some of the most popular visualization software to be used in a Linux based environment [11]. Nevertheless, it is important to highlight that, nowadays, several companies and developers decided to equip their simulation tools (such as Fluent) of advanced features, which allow to generate images for the graphical visualization of data in an automatic and run-time manner. Therefore, if the user application is based on such kind of software, the task of the *VisualGrid Producer* module will extend such functionalities through the registration on the Grid of the images so produced. In order to enable their correct recover during the processing phase, the images distribution of each monitored application is recorded on the LFC Catalog under the logical path */grid/VO NAME/VisualGrid*. More precisely, for each user application, the *VisualGrid Producer* module creates a directory whose name is an unambiguous ID. Such directory is then used to create symbolic links to the images (supported formats are "png","jpg") stored as individual files on the SEs.

The *Images Streaming* activity is performed by the *VisualGrid Client* module and one or more *VisualGrid Jobs*. Running on the user's device, the *VisualGrid Client* module offers a GUI by which the user can specify the application that has to be monitored, and other parameters such as images format, video frame rate, video resolution. Task of this module is to submit the *VisualGrid Jobs* to the Grid infrastructure. Executed in worker nodes (*VisualGrid Worker Nodes*) usually different from the *Application Worker Node*, they contain all the software needed for the creation and the streaming of video files, housing information about the monitored application. In this way, it is not necessary a pre-installation of the requested programs on the WNs, with the consequence of an increasing of the Input Sandboxes size.

Once the user submit the monitoring request, the *VisualGrid Client* module verifies the number of available images, by using the LFC Catalog described above, and launches an appropriately number of *VisualGrid Jobs*. Each *VisualGrid Job* will then recover a subset of the images produced up to that point and, carried out the required encoding task, it will activate the streaming server waiting for a request from the client player. The free command line video decoding, encoding and filtering tool MEncoder was used [12] to generate multimedia contents from one or more JPEG, PNG or other image files, customizing them according to user's desires [12]. In order to drive the player to request the streaming in a correct order, the *VisualGrid Client* module generates a playlist containing the proper association of each *VisualGrid Job* to the correct video chunk. Each playlist element is a HTTP GET request like: *http://unime-wn-35.me.pi2s2.it/stream?chunk=video1.avi* where the URL contains the address of the WN that hosts the *VisualGrid Job* and the name of the video's chunk.

In the *Images Streaming* activity, a strong interaction between the *VisualGrid Client* module and *VisualGrid Jobs* is required to obtain a correct workflow of the application. In fact, the former has to know which worker nodes on the Grid are used for the creation and the streaming of the video. The solution adopted in *VisualGrid* is to allow the communication between such components by using of a messages oriented middleware based on the Java Messaging Service (JMS) [13]. A publish/subscribe model [14] has been used to enable *VisualGrid Jobs* to post information about their execution status and adequately manage the application flow. More specifically, *VisualGrid Jobs* can send two kind of JMS messages:

- *READY* used when video file has been correctly created, and *VisualGrid Job* is ready for the streaming;
- *FAILED* used if something went wrong during the encoding phase.

An example of the application workflow is depicted in the diagram of Fig. 3. In the simplest scenario, where a single *VisualGrid Job* is used, we assume that the user submits a job containing all that is necessary to execute his application together with the *VisualGrid*

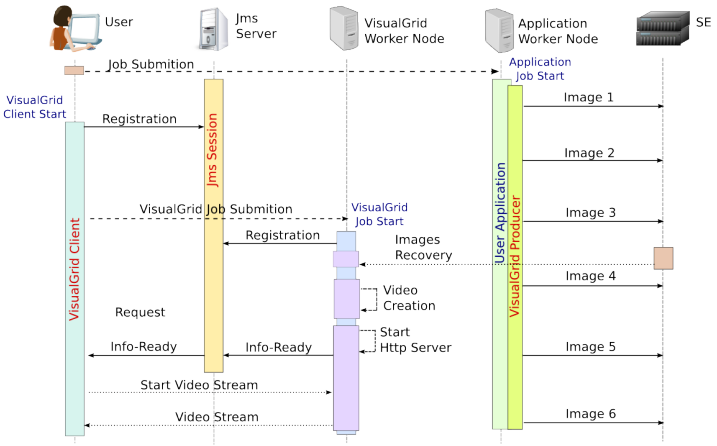


Fig. 3. A typical application flow diagram

Producer module. At regular intervals, the *VisualGrid Producer* creates an image representing the application status and records it on the Grid SEs. Whenever the user wishes to monitor his application, he run the *VisualGrid Client* module on his device. This module automatically executes the registration on the messaging server and waits for a request by the user. Once received the monitoring request of the user, it launches a *VisualGrid Job* that recovers all the available images and encodes them into a video file. End the video creation phase, the streaming server is started and the *VisualGrid Job* sends a *READY* message to the *VisualGrid Client* module. Finally, it will wait until the player requests the streaming of its video and, completed the streaming, it will stop.

V. Experimental Results

Wireless sensor networks are application-specific networks used to monitor specific areas by collecting several measures of physical parameters, sampled at different space and time intervals. Such networks consist of a large number of tiny sensor nodes deployed in a given region. Typical applications are environmental monitoring, industrial sensing, battlefield awareness and so on. The sensing activity of each node has to be correlated with a communication task in order to send the measured values to one or more central elaboration systems (sinks). Very limited resources in each sensor node and the high cost of radio transmission in terms of energy consumption imply that routing signalling has to be carefully managed in order to increase the lifetime of the whole network. Therefore, scientific researchers have oriented their efforts to the implementation of more effective routing protocols able to overcome these issues.

The use of a simulator to analyse a routing protocol for wireless sensor networks [15] has been exploited in order to test the performance of the *VisualGrid* tool. The analysis was performed using the OMNET++ simulation framework [16]. Several simulations, rendering different scenarios, were executed in remote on the Grid infrastructure. At intervals of 1 minute, an image (see Fig 4) - providing a graphical representation of the size indicated by the researcher as parameter to be monitored ("pheromone" level distribution node by node) - was produced using the common software Gnuplot [17]. JPEG images of 3200x2000 pixels, dimension of about 175 Kbytes, stored on Grid as individual files has been used. While the simulations running, *VisualGrid* has been exploited by the researcher to require the video composed

by sequential images in order to verify the temporal evolution of the selected parameter ("pheromone" level in the network) and the convergence of the protocol. The video file has been encoded with the following parameters: dimension 1024x768 pixel, frame rate of 1fps, lavc as video codec.

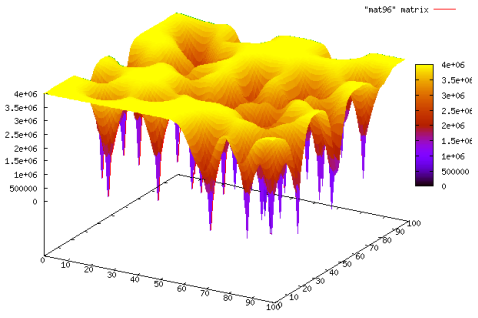


Fig. 4. Example of image produced during experimental analysis concerned to simulations for wireless sensor network

In term of user satisfaction, the most important parameter that has been evaluated is the *vision latency time* (T_{VL}), expressed as the time interval between the submission of monitoring request by the user and the video streaming execution. Referred to the time needed for *"N" VisualGrid Jobs* to be ready for the streaming, this parameter is strongly affected from both the time needed to have all jobs running on Grid (*scheduling time* T_S) and the interval (*processing time* T_P) needed for each of them to recover the images, to encode the video file and to start the streaming server

As indicated in a previous work [18], T_S depends on the load of the Grid infrastructure and is not predictable. Characterized by a standard deviation equal to the 103% of its mean and a maximum value almost nine times the mean itself, it often forces users to wait a long period before resources become available. A solution based on a sort of jobs reservation, for example by mean of idle_jobs, can be used in order to improve the perceived quality of service [18].

The curves of Fig. 5, instead, show the

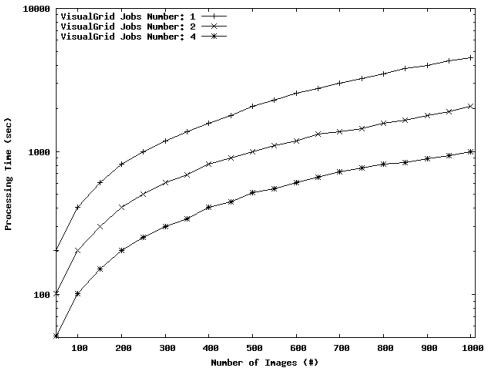


Fig. 5. The variation of T_P versus the number of images for several numbers of exploited *VisualGrid Jobs*

variation of the processing time (T_P) versus the image number for several numbers of exploited *VisualGrid Jobs*. As expected, the results demonstrate that with an increasing number of images, the processing time increases in a linear manner. Similarly, with an equal image number it decreases as the number of *VisualGrid Jobs* increases.

Finally, as shown in Fig 6, experimental analysis also highlights as the time needed for each single job to recover images (T_R) represents about 98,77% of its entire processing time T_P. This mainly depends on authentication and authorization mechanisms implemented in Grid and activated each time a user or job tries to access a data file.

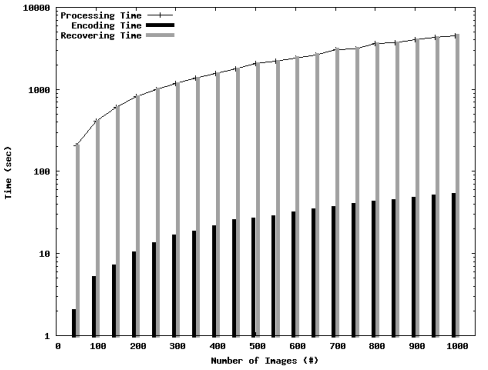


Fig. 6. Contributions of T_R, T_E for T_P of each *VisualGrid Job*

VI. Conclusions

One of the most important drawbacks of current middleware is the inability for users to monitor the evolution of their applications. The implemented monitoring facilities provide them by the system status of their applications (such as scheduled, running, finished or aborted). But, this kind of information is often not enough for users that need of more specific information. In this paper, the implementation of *VisualGrid* has been presented. It represents a valid approach for solution the problem. Based on scientific visualization and streaming techniques, it enable users of a gLite based Grid system to monitor the evolution of their applications. Such feature may be very useful, especially for applications characterized by a long term execution. Feasibility has been proved by evaluating the performance of our *VisualGrid* tool.

Acknowledgment

This work makes use of results produced by the PI2S2 project managed by the Consorzio COMETA. A project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione (PON 2000-2006). More information is available at www.consorzio-cometa.it and www.pi2s2.it.

We also would like to thank Dr. G. Andronico for the useful preliminary discussion on the topics covered in this paper.

References

[1] V. Kalyaev and A. Kryukov, "Problem of application monitoring in grid systems," in *Proceedings of the XIX International Symposium on Nuclear Electronics and Computing 2003, Bulgaria, Varna*, 2003.
[2] I. Foster, C. Kesselman, and S. Tuecke, "The anatomy of the grid: Enabling scalable virtual organizations," *International J. Supercomputer Applications*, 2001.
[3] M. Montes, J. G. Callejon, R. Pollan, and M. R. del Solar, "Management of a grid infrastructure in glite with virtualization," in *In Proocedings of the 1st Iberian Grid Infrastucture Conference*, 2007.
[4] "gLite middleware - http://glite.web.cern.ch/glite/."
[5] "Egee project - http://public.eu-egee.org/."
[6] "Condor project - http://www.cs.wisc.edu/condor/."
[7] "Globus toolkit - http://www.globus.org/toolkit/."
[8] "Lcg project - http://lcg.web.cern.ch/lcg/."
[9] "Vdt - http://vdt.cs.wisc.edu/components/vdt.html."
[10] R. Raman, M. Livny, and M. H. Solomon, "Matchmaking: Distributed resource management for high throughput computing," in *Proceedings 7th International Symposium on High Performance Distributed Computing*, 1998, pp. 140–146.
[11] M. Jones, "Data visualization tools for linux," IBM DeveloperWork, Tech. Rep., 2006.
[12] "Mencoder - http://www.mplayerhq.hu/."
[13] "Jms - http://java.sun.com/products/jms/."
[14] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Computer Surveys*, vol. 35, no. 2, pp. 114–131, 2003.
[15] M. Paone, L. Paladina, D. Bruneo, and A. P. A, "Swarm-based routing protocol for wireless sensor," in *IEEE International Symposium on Network Computing and Applications (IEEE NCA07) 2007*.
[16] "Omnet++ - http://www.omnetpp.org/."
[17] "Gnuplot - http://www.gnuplot.info/."
[18] D. Bruneo, G. Iellamo, G. Minutoli, and A. Puliafito, "Gridvideo: A practical example of non-scientific application on the grid," *IEEE Transactions on Knowledge and Data Engineering*, 2008, accepted for future publication.

G. Minutoli received the Laurea degree in Electronic Engineering from the University of Messina, Italy, in 2005. He is currently attending his Ph.D in Advanced Technologies for Information Engineering at the University of Messina. His research interests are in the areas of distributed computing, Grid, virtualization.

M. Paone was born in Messina on March 20th 1973. He received his Degree in Electronic Engineering from the Engineering Faculty of the University of Messina (Italy) in 2003. In December 2006, he received the Ph.D. degree in Advanced Technologies for Information Engineering from the University of Messina. Until November 2008, he worked with the Consorzio COMETA research group at University of Messina. Scientific activity of Dr. Paone has been focused on studying mobile networks systems, and Quality of Service (QoS) management in wireless networks and ad-hoc networks.

M. Fazio was born in Messina on July 17th, 1976. She received her Degree in Electronic Engineering from the Engineering Faculty of the University of Messina (Italy) in 2002, the Ph.D. degree in 2006 in Advanced Technologies for Information Engineering at the University of Messina, and the Post-Doctoral degree in 2007. Scientific activity of Dr. Fazio has been focused on studying networks systems. Her main research interests include ad hoc networks.

G. Iellamo was born in Messina December 1st 1979. In May 2005 he received the master degree in Electronic Engineering from the University of Messina, final score 110/110 summa cum laudae. Until November 2008, he worked with the Consorzio COMETA research group at University of Messina. Now he is an IT-consultant for a major TLC company. His main interests include Distributed Computing and Grid Systems

A. Puliafito is a full professor of computer engineering at the University of Messina, Italy. His interests include parallel and distributed systems, networking, wireless and GRID computing. He is currently the director of the RFIDLab, a research lab on RFID and wireless, the vice-president of the Consorzio Cometa whose aim is to enhance and exploit high performance computing. He is actively working in grid computing and virtualization in the projects TriGrid VL, PI2S2 and RESERVOIR.

Automatic Antennas Design by means of Genetic Programming

G.A. Casula[1], G. Mazzarella[1] and N. Sirena[2]

[1]*Dip. di Ing. Elettrica ed Elettronica, Piazza d'Armi, Cagliari, Italy, {a.casula,mazzarella}@diee.unica.it*

[2]*Consorzio COSMOLAB, via Marengo 3, Cagliari, Italy, nicosirena@gmail.com*

***Abstract*—Genetic optimization has been used in the last years for solving different electromagnetic problems. Genetic optimization assumes a fixed structure from the beginning, while Genetic Programming determines the antenna shape and dimensions as an outcome of the procedure. In this work we use genetic programming to design wire antennas. The performances of each antenna generated by the genetic programming during the optimization process are evaluated by a method of moments code, NEC-2.**

***Index Terms*— Cybersar, Genetic Programming, Automatic Design, Antenna.**

I. INTRODUCTION

TRADITONAL antenna design techniques require to the designer continuous intervention for the evaluation of the intermediate results, and knowledge and experience in the field to effectively change the structure under development. The alternative we propose allows to automate the antennas design by means of Genetic Programming, obtaining performances comparable (even greater) with standard methods (Yagi or LPDA), and original solutions not easily reachable with traditional techniques.

Genetic optimization [1] has been proposed for the solution of electromagnetic problems and has found many successful applications in the last years. When Genetic Programming approaches [2] have been proposed for small wire antenna design [3], the full capability of evolution-inspired methods has been exploited. Genetic Programming determines the antenna shape and dimensions as an outcome of the procedure.

Genetic Programming mimics the behavior of the natural evolution in the search of the individual showing the best adaptation to the local environment. Genetic Programming starts with an initial population of randomly generated computer programs composed of functions and terminals appropriate to the problem domain. The functions may be standard arithmetic operations, standard programming operations, mathematical or logical functions, or even domain-specific functions.

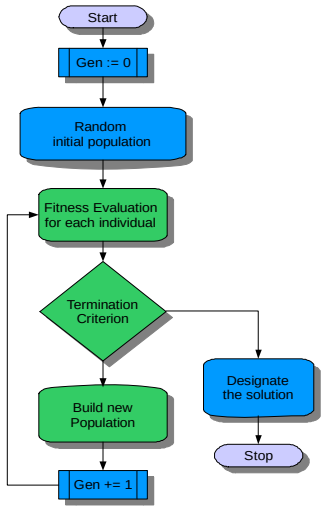


Fig. 1. Flow-Chart of an evolutionary run.

The main (meta)-operators used in genetic programming are:

- Crossover: switching one subtree of a individual with a randomly selected subtree from another individual in the population;
- Mutation: modifies a whole node in the selected individual

The performance of each in the population is measured by its "fitness".

In the design process, each individual in the population represents an admissible antenna. The performances of each antenna are evaluated by NEC-2, a method of moments code.

Genetic Programming optimization process starts with a random initial population, and it may move in different directions, converging to different results for the final designed antenna. Genetic Programming start point is totally random: it is a global optimization method, particularly suitable for solving problems with several parameters and no clear starting location in the solution space. As a result, it can lead to particular configurations for the design antennas, which are not achievable using standard design techniques.

A fitness, based on how closely the design meets the desired performances, is associated to each antenna configuration. Individuals with the better values of the fitness function get a larger probability of breeding. New populations are generated and evaluated until suitable stopping rules are met.

The fitness we developed is able to direct the evolution process on a structure with reduced size, with the highest end-fire gain, and with an input match as better as possible in the widest frequency range. The form of the fitness function can be a critical point, since only a suitable fitness can lead the optimization process to significant results.

The solution space of a GP design is very large, thus the problem can be heavily ill-posed. We require a low VSWR over all the required bandwidth since it is needed to use the antenna. Actually, being a near-field requirement, its inclusion also stabilizes the problem, without additional cost. Moreover, NEC-2 takes into account even the conductor losses, allowing to prevent from superdirective solutions.

The resulting fitness function is composed by three main terms: the first term (FM) takes into account the input matching of the antenna, the second term (FG) takes into account the antenna gain including the effect of conductor losses, and the last term (FS) takes into account the size of the structure. The fitness can be expressed as:

$$Fitness = \left(F_M + F_G\right) \cdot F_S \quad (1)$$

where:

$$\left.\begin{aligned} F_M &= \left|1 - \overline{SWR}\right| \cdot \alpha_M \\ F_G &= \left|\frac{G_{MAX}}{\overline{G}}\right| \cdot \alpha_G \\ F_S &= 1 + \frac{D_{REAL} - D_{REQ}}{D_{REQ}} \cdot \alpha_S \end{aligned}\right\} \quad (2)$$

In (2) α_M, α_G and α_S are suitable weights, while $\overline{SWR}$ and $\overline{G}$ are the mean values of SWR and gain over the bandwidth of interest, D_{REAL} represents the real antenna size and D_{REQ} is the requested size for the antenna.

II. Grid Implementation

A grid implementation has been done with a Master-Slave architecture. The master, which runs in the User Interface, build the individuals of each generations and the slaves have the task of evaluating them. Each slave is submitted in grid as a job and, once started,

opens a TCP/IP connection to exchange data with the master.

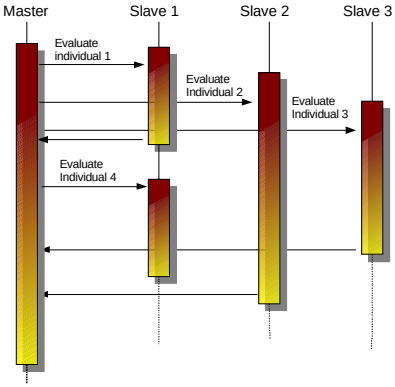


Fig. 2. Master-Slave Interaction.

A queue (where some individuals are inserted, waiting for their evaluation) is associated with each slave, and when a slave finishes an evaluation a new individual is inserted in his queue. This way the evaluation process (i.e. the most expensive one) can be divided on N machines, obtaining a performance increase proportional to N. The master, using Java API, is able to automatically submit new jobs/slaves and/or terminate the ones in execution, so the user acting on specific controls of the GUI can dynamically vary the number of active slaves.

III. Results

The automated design using genetic programming has been applied to several antennas, with different sizes, number of elements, and operation frequencies, and with different requirements both on Gain and input matching, giving very good results. In this section we present a wire antenna obtained with our technique. Its performances have been compared with the performances of two traditional Yagi; one of comparable size and and one of comparable performances. The operation frequency is 500 MHz, and the requested bandwidth is at least 60 MHz (i.e. 12%, from 470 MHz to 530 MHz, the typical bandwidth of a standard Yagi). The antenna size is very small, since it fits in a box large $0.91\lambda_0$x $0.16\lambda_0$x $0.99\lambda_0$, being λ_0 the space wavelength at the operation frequency of 500 MHz. Its SWR is less than 2 in a bandwidth of 110 MHz, and its gain inside this bandwidth is comprised between 14 and 26 dB.

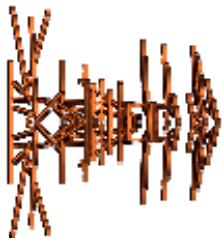


Fig. 3. Designed Antenna Structure

The Yagi with comparable size fit in a box large $1.06\lambda_0$x $0.002\lambda_0$x $0.435\lambda_0$, that with comparable performances fit in a box large $21.75\lambda_0$x $0.002\lambda_0$x $0.435\lambda_0$.

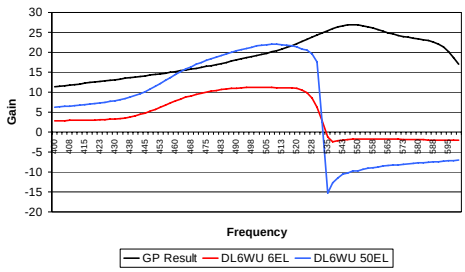


Fig. 4. Gain comparison between the Genetic Designed Antenna and the two Yagis.

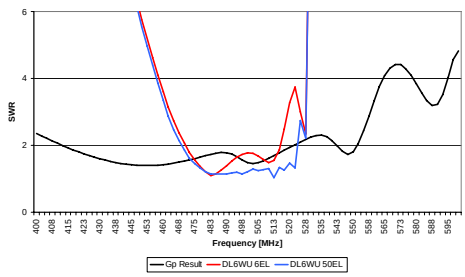


Fig. 5. Gain comparison between the Genetic Designed

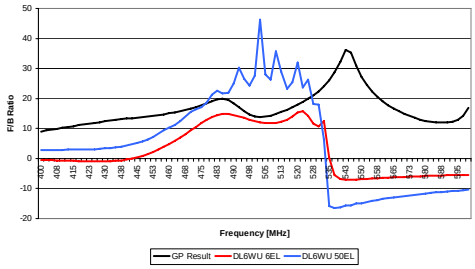


Fig. 6. F/B comparison between the Genetic Designed Antenna and the two Yagis.

IV. Conclusions

An automated design procedure for wire antennas, based on Genetic Programming concept, have been devised. Its main advantage is the ability to explore a far larger solution space than standard optimization algorithms. The presented results show that a wire antenna with high gain, reduced size and a large bandwidth can be designed, and its performances are robust with respect to tolerances or atmospheric events.

Acknowledgments

This work makes use of results produced by the Cybersar Project managed by the Consorzio COSMOLAB, a project co-funded by the Italian Ministry of University and Research (MIUR) within the <<Programma Operativo Nazionale 2000-2006 "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" per le Regioni Italiane dell'Obiettivo 1 (Campania, Calabria, Puglia, Basilicata, Sicilia, Sardegna) – Asse II, Misura II.2 "Società dell'Informazione", Azione a "Sistemi di calcolo e simulazione ad alte prestazioni">>. More information is available at http://www.cybersar.com.

References

[1] M. Mitchell, "An introduction to Genetic Algorithms", Mit Press, 1999.
[2] J.R. Koza, "Genetic Programming: On the Programming of Computers by Means of Natural Selection", MIT Press, 1992.
[3] J.R. Koza, M.A. Keane, M.J. Streeter, W. Mydlowec, J. Yu, G.Lanza, "Genetic Programming IV: Routine Human-competitive Machine Intelligence", Springer, 2003.

Giovanni Andrea Casula was born in Sassari, in 1974. He received the Laurea Degree summa cum laude in Electronic Engineering in 2000 and the Ph.D. in Electronic Engineering and Computer Science in 2004 from the Università di Cagliari. Since March 2006 he is Assistant Professor of Electromagnetic Field and Microwave Engineering at the Department of Electrical and Electronic Engineering of the Università di Cagliari. His current research interests are in the field of antenna pattern synthesis, analysis and design of patch and slot antennas.

Giuseppe Mazzarella (M'85–SM'90) was born in Napoli, in 1959. He received the Laurea Degree in Electronic Engineering summa cum laude in 1984 and the Ph.D. in Electronic Engineering and Computer Science in 1989 from the Università Federico II in Napoli. From 1990 to 1992 he was assistant professor in the department of Electronic Engineering of the Università Federico II in Napoli. In 1992 he joined the Department of Electrical and Electronic Engineering of the Università di Cagliari where he is now full professor of Electromagnetic Fields and Antennas theory and design. His current research interests are in the field of antenna pattern synthesis, analysis and design of patch and slot antennas and inverse problems in electromagnetics.

Nicola Sirena was born in Cagliari, in 1972. He received the Laurea Degree in Electronic Engineering in 2007 from the Università di Cagliari. Since June 2007 he works in collaboration with the Department of Electrical and Electronic Engineering of the Università di Cagliari. His current research interests are in the field of automated antenna design.

Electromagnetic Field Strength Prediction in Urban Scenario by using a Grid Environment

S. Coco[1], A. Laudani[1] and G. Pollicino[1]
[1]*Dipartimento di Ingegneria Elettrica, Elettronica e dei Sistemi (DIEES), University of Catania, V.le A. Doria 6, Catania, Italy*

***Abstract*—A Grid-based 3-D ray tracing procedure is presented for the prediction of electromagnetic field in urban environment. In the procedure electromagnetic field strength in the presence of multipaths, reflections and diffractions is evaluated by exploiting the ray tracing method in Grid environment. The obtained results show the good performance of the Grid environment for the prediction leading to reduction of one-order of magnitude of the computing time.**

***Index Terms*—Grid Computing, Ray-tracing, Electromagnetic Field Prediction.**

I. Introduction

A CRUCIAL objective of wireless communication systems is to assure adequate covering within the interesting region keeping electromagnetic (EM) field levels within the bounds allowed by regulations. Among the several methods currently adopted for the evaluation of environmental EM field strength the Ray-Tracing technique is popular for accurate predictions in urban scenarios or indoor environment, in presence of obstacles, such as buildings, for which it is not possible to use simplified EM field propagation models. In fact, differently from radio communication direct links, which work under LOS (Line Of Sight) conditions, for wireless communications in urban scenarios, the propagation conditions are NLOS (No Line Of Sight), and thus a more sophisticated approach is necessary for EMF strength evaluations. The ray tracing technique, which is the most popular technique for the design and the development of mobile networks, satisfies this need [1], [2], [4], [3], [5]. Although Ray-tracing propagation models are commonly used for the prediction of the field strength in wireless mobile communication networks, these models suffer from long computation times. For large areas in urban or indoor complex scenarios, the computation times can be in the range of days. Several solutions have been studied in order to solve this kind of problems, but only the possibility to adopt distributed computing can lead to realistic simulations of real urban environment within a reasonable computing time [7]. In this paper we present a GRID computing [6] based Ray-Tracing prediction of electromagnetic fields in urban scenarios. The new model has been set up in the GRID environment in the framework of the PI2S2 project [8], by using the know-how, earned during the development of specialized codes for the prediction of environmental electromagnetic fields. We individuate and implemented a set of procedures able to obtain a remarkable reduction of the computing time. The paper is structured as follows: in the section II a short description of the ray tracing algorithm for the construction of EM field strength map is presented in order to underline the aspects important for a parallel implementation; in the section III the distributed computing approach developed and the results obtained are reported.

II. 3-D Ray-Tracing Prediction

For the prediction of the EM field strength emitted by radio-base transmission systems a ray optical method (ray tracing) is employed in the presence of multipaths, reflections, and diffractions. The ray tracing technique is the most popular technique for the design and the development of mobile communication networks and is based on optical geometry (GO) and on its generalization, the geometric theory of diffraction (GTD). The two main assumptions of GTD are that all the obstacles must have size larger than wavelength and that reflections can be described by plane wave formulas. The construction of EM field strength map by means of the ray-tracing algorithm consists of two main phases: in the first the visibility between elements of the environment are represented by a visibility tree; in the second phase all the propagation multipaths between a transmitter T and a receiver R (measurement point) are determined and the contributions of these multipaths to the total field strength are computed. The propagation environment is modelled by means of a set of object points O_k, belonging to the various obstacles. These points can be of two different kinds, diffraction points (O_{kd}), i.e. points lying on the edge of the obstacle, and reflections points (O_{kr}), i.e. points lying on the face of the obstacle. The diffraction points contribute to the total field by adding a diffracted component computed according to specific formulas, while the reflected points contribute with the reflected field. To determine the field intensity due to each path we consider the concept of propagation path of order N between a transmitter T and a receiver R, defined as an ordered sequence of points

$T, O_k[1], .., O_k[N], R$

where $O_k[i]$ is a generic object point, and each couple of consecutive elements in the preceding sequence is directly visible, i.e. there is no obstacle between them. In order to reduce the time needed for the construction of all the possible paths between T and R (considering also that usually we have to deal with many measurement points), it is possible to use a structure, "the visibility tree", which contain in each branch a propagation path. In this way the complete search of all the admissible path can be easily obtained by visiting the visibility tree. The visibility tree consists of a certain number of levels corresponding to the maximum number of interactions between object points to be considered in the ray optical propagation model. At the first level there is only the root represented by the transmitter T. The branches originated from this root T constitute the second level, to which all the object points O_k directly visible from T belong. In this second level each object point O_k represents the root for the successive level of the tree constituted by all the object points O_j directly visible from O_k. In this way in the branches of the each levels it is possible to find all the object points directly visible from object points belonging to the preceding level, and the connections are in such a way to specify all the admissible paths from T to each object points. The construction of the tree can be made recursively and the only condition to check in the extraction of admissible path from the visibility tree is that each O_k be present only once for each propagation path. The determination of the direct visibility is performed by using simple geometrical concepts such as intersection of a line with a face and a plane, which represent obstacles. Even in this case a pre-processing of the environment, excluding some obstacles according to their position, allows a reduction of the computational effort. Once the visibility tree is built, the field strength in the measurement points P_i is obtained by adding the direct component due to transmitter T (only if T and P_i are directly visible), the reflected components, due to all the reflection object points O_{kr} directly visible from P_i, and the diffracted components, due to all the diffraction points O_{kd} directly visible from P_i. Clearly the higher is the order of the propagation path, that is the number of levels of visibility tree, the higher is the computational effort to evaluate each field component. Usually a tree of three level is a good trade-off between computational cost and a sufficiently

accurate description of the field strength map. For the computation of the various field contribution (reflected/transmitted/diffracted/scattered from obstacles, etc) appositely expression are used, considering the characteristic parameters of the material constituting the obstacle [2], [4], [5]. From this description is possible to individuate various methodologies of segmentation of the computation. For example it is possible, when there is more than a source, to split the analysis considering a visibility tree for each source. In addition, the domain under analysis can be subdivided in sub-regions, for example according to the height from the ground, or to any other suitable discretization. In this case it is important to note that the construction of the visibility tree relative to a source cannot be usefully split since this could lead to cancel some important propagation paths. According to these observations in the next section the strategies implemented in the GRID environment will be described.

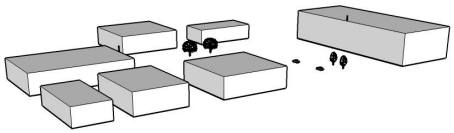


Fig. 1. 3-D view of the analyzed example

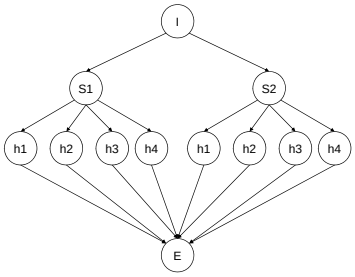


Fig. 2. The DAG representing the considered scenario

III. GRID-BASED EMF PREDICTION

In order to submit the desired task to the GRID infrastructure it is necessary to use an interface language able to pass the required information relative to the various jobs. The Job Description Language (JDL) is a high-level, user-oriented language for describing jobs and aggregates of jobs, by means of peculiar attributes, which represent request-specific information and specify in some way actions that have to be performed by GRID structure. For example if an electromagnetic analysis can be split for different values of frequency it is possible to adopt a so-called parametric job, which contains parametric attributes (regarding the arguments of the simulation program), whose values can be varied according to another attribute also specified in the job description. In our case a parametric job could be useful, but following this approach a post processing phase is necessary. Consequently we adopted an aggregate of jobs called DAG. A Directed Acyclic Graph (DAG) represents a set of jobs having a sort of dependence among themselves. Practically the jobs can be considered as the nodes in a directed acyclic graph and the arcs identify the dependencies.

In our case a DAG JDL has been setup in order to automatically submit the Ray Tracing prediction In particular a first job (from which all the others depend) reads all the necessary data and subdivides the problem into sub-jobs according to the specific analysis. If only a source is present the subdivision is in terms of domain, as discussed before, while if more than a source is considered in the analysis a first subdivision regards the sources and each jobs perform a further subdivision in terms of domain. Then all the jobs results are collected by an end node, which restitutes them in such a way that all this procedure is transparent to the user. In this way the computation cost of a ray tracing prediction can be noticeably reduced and the results can soon be made available. In order to better explain the procedure developed an example of EMF rey-tracing prediction in urban scenario is hereafter presented. The environment is represented in the 3D view in fig. 1.

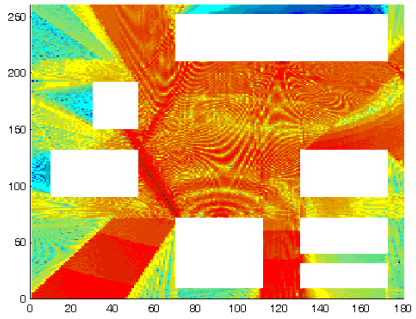


Fig. 3. 2-D view of electromagnetic field strength in the considered urban environment (z=1.5 m).

It is a typical urban area of 260 m 180 m, containing 7 buildings, whose height varies between 10 and 20 m. Two sources are present in correspondence of the roof of two of the buildings. The analysis has been carried out considering both the sources and a discretization step of one meter in the x, y axes and respectively the z-level of 1.5, 2.0, 5.0 and 15 m.

The analysis has been split according the DAG represented in fig. 2. The node I reads and processes the input data of the problem and launches the job S1 and S2, which have the role to subdivided the analysis for the two sources and to launch the job indicated by nodes hi, which perform the field strength computation for the differ quotes z (1.5, 2, 5 and 15). In this way this computation which required from 4 to 8 hour according to the level of visibility tree used in a PC, can be performed in about 30 minutes in the GRID environment. Fig. 3 shows the electromagnetic field strength obtained for z = 1.5 m. The obtained results have shown the good performance of the GRID environment for the prediction leading to reduction of one-order of magnitude of the computing time.

ACKNOWLEDGMENT

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.).

REFERENCES

[1] T. K. Sarkar, el al., *A survey of various propagation models for mobile communication*, IEEE Antennas and Propagation Magazine, Vol. 45, issue 3, June 2003, pp. 51-82.

[2] Salvatore Coco, Antonino Laudani, and Letizia Mazzurco, *A Novel 2-D Ray Tracing Procedure for the Localization of EM Field Sources in Urban Environment*, IEEE Transactions on Magnetics, vol. 40, no. 2, pp. 1132-1135, March 2004.

[3] F.M. Landstorfer, *Wave Propagation Models for the Planning of Mobile Comunication Network*, European Microwave Week 1999, Munich (Germany), Oct. 1999.

[4] S.Y. Tan and H.S. Tan, *A microcellular communications propagation model based on the uniform theory of diffraction and multiple image theory*, IEEE Transactions on Antennas and Propagation, Volume: 44 Issue: 10 , Oct. 1996 pp. 1317-1326.

[5] K.R. Schaubach; N.J. Davis and T.S. Rappaport, *A ray tracing method for predicting path loss and delay spread in microcellular environments*, 42nd IEEE Vehicular Technology Conference, 1992, 10-13 May 1992, vol. 2, pp. 932 -935.

[6] Salvatore Coco, Antonino Laudani, and Giuseppe Pollicino, *GRID-Based Prediction of Electromagnetic Fields in Urban Environment*, IEEE Transactions on Magnetics, Vol. 45, No. 3, March 2009, pp. 1060-1063.

[7] I. Foster, C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, 1999.

[8] PS2I2 - http://www.pi2s2.it and http://www.consorzio-cometa.it. (28 February 2009).

Multi-objective Design Space Exploration of Embedded Systems in a Grid Environment

V. Catania[1], G. De Francisci Morales[1], A. G. Di Nuovo[1], M. Palesi[1] and D. Patti[1]
[1]Dipartimento Ingegneria Informatica e delle Telecomunicazioni (DIIT)
University of Catania, Italy
{vcatania,gmorales,adinuovo,mpalesi,dpatti}@diit.unict.it

***Abstract*—The embedded systems market is without any doubt the largest and most significant application area for microprocessors. In this work we assess the use of Grid-based environments in Design Space Exploration (DSE) of a complex highly parameterized VLIW-based System-on-Chip. The free parameters which characterize such system span a design space of over 10^{13} system configurations so that even considering an evaluation time of a few seconds for each configuration, exhaustive exploration would take hundreds of years on single machine. To the best of our knowledge, all the techniques proposed in literature for such systems rely on the use of a computational infrastructure which is mono-processor in nature. The main contribution of this work is to demonstrate the use of Grid and High Performance Computing (HPC) environments as a viable solution to tackle the complexity of multi-objective optimization of embedded systems. We believe that the availability of Grid computing infrastructures should be exploited and analyzed in the context of DSE to cope with the ever growing increase in complexity of current and next generation SoC platforms. The presented results show a significant reduction of computing time nedded to perform an exploration and introduce some new challenges that an embedded systems designer has to face when using Grid and HPC environments, such as the impact of communication overhead and the scalability of a multi-objective DSE strategy.**

***Index Terms*—VLIW, Embedded Systems, Genetic Algorithms, Grid.**

I. Introduction

THE increasing complexity of modern electronic systems, together with increasing time-to-market pressure, has led the embedded system industry to adopt pre-designed System-on-a-Chip (SoC) architectures known as *SoC platforms* [1], [2], [3]. From one hand a SoC platform has to be general enough to be used in different application scenarios, so as to offset the design, verification and manufacturing costs involved. On the other hand, it has to be specific in order to meet the typical design constraints like silicon area, performance, and power dissipation.

A SoC platform is typically made up of a number of parameterized intellectual property (IP) cores communicating via an interconnection system which is also parametric. Each parameter has an associated range of discrete values within which it can vary. When a value is set for each parameter the system is said to have been configured. As the number of possible configurations that can be mapped on a parametric system is equal to the product of the cardinalities of the ranges of variation for each parameter, the size of the space of configurations grows exponentially as the number of parameters increases. Exhaustively exploring the design space looking for the best system configuration for a given application (or, more in general, for a set of applications) is computationally infeasible in practical cases. In fact, each visited system configuration must be

evaluated by means of a system-level simulation which typical is a computational intensive task.

It is therefore of fundamental importance to define exploration methods that can combine two basic but unfortunately conflicting requirements: efficiency and accuracy. Efficiency is a measure of the time required to complete an exploration. Accuracy is an index of quality of the solutions obtained. The capacity to find a good trade-off between these requirements is the challenge now facing Design Space Exploration (DSE) methods.

Another important difficulty in this context, is the fact that the search for optimal configurations is a matter of multi-objective optimization, where some of the objectives conflict with others, for example performance and power consumption. Although this causes a considerable increase in the complexity of DSE strategies, it has the advantage of offering the SoC designer not one but a set of optimal configurations (Pareto optimal set) from which he/she can choose the one that represents the best trade-off in relation to the set of constraints which have to be met.

Typically, the DSE techniques proposed in literature tackle the DSE problem by means of either exact [4] or approximated [5], [6] approaches. In [4] Givargis *et al.* propose an exact technique based on the notion of dependence between parameters. The basic idea of their approach is to cluster dependent parameters and then carry out an exhaustive exploration within these clusters. If the size of these clusters increases too much due to great dependency between the parameters, the approach becomes a purely exhaustive search, with a consequent loss of efficiency. To deal with these problems approximate approaches have been proposed in literature which further reduce the exploration space but give no guarantee that the solutions found will be optimal. Fornaciari *et al.* in [5] use sensitivity analysis to reduce the space of exploration from the product of the cardinalities of the sets of variation of the parameters to their sum. This approach can be seen as a simplified version of [4], as all the parameters are considered to be independent. Another approximate approach was proposed by Ascia *et al.* in [6]. They propose the use of multi-objective genetic algorithms as optimization technique for DSE. Other DSE approaches which perform the pruning of the design space con be found in [7], [8], [9], [10]. Most of the approaches belonging to the second category are of limited applicability and not general (or scalable) since they are often tailored for a specific system architecture. The use of an analytical model to speed up evaluation of a system configuration is presented in [11]. Statistical simulation is used in [12] to enable quick and accurate design decisions in the early stages of computer design, at the processor and system levels. A recent approach [13] uses statistical simulation to speed up the evaluation of configurations by a multi-objective genetic algorithm. In [14] it is shown that the use of an intelligent Multi-Objective Evolutionary Algorithm (MOEA) reduces the exploration time by two orders of magnitude respect to the classical MOEA, without any loss in accuracy of results.

In this paper we present a case study of design space exploration of a complex highly parameterized VLIW based SoC platform. The 18 free parameters which characterize the platform span a design space of over 10^13 system configurations. Even considering an evaluation time of a few seconds for each configuration, exhaustive exploration would take hundreds of years on single machine. We test the use of High Performance Computing (HPC) as a viable solution to tackle with DSE related problems. For the best of our knowledge, all the DSE techniques proposed in literature relies on the use of a computation infrastructure which is typical mono-processor in nature. We believe that the availability of distributed or grid computational infrastructures should be exploited and analyzed in the context of DSE to cope with the ever growing increase in complexity of current and next generation SoC platforms. It should be pointed out that HPC is not as cheap as statistical or machine learning approaches (e.g. an HPC machine, like the one we used

for this work, costs about 50 thousands euros).

This paper is organized as follows. Section II present a simulation environment, called Epic-Explorer, and in Section III we describe the work done to integrate the Epic-Explorer into an HPC environment. In Section IV the methodology is applied to a real case study. Finally Section V summarizes our contribution and outlines some directions for future work.

II. Simulation Environment

To evaluate and compare the performance indexes of different architectures for a specific application, one needs to simulate the architecture running the code of the application. In this work we use the Epic Explorer [15] simulation environment. To make architectural exploration possible both the compiler and the simulator have to be retargetable. Trimaran [16] provides these tools and thus represents the pillar of the Epic Explorer. Epic Explorer is an environment that not only allows us to evaluate any instance of a platform in terms of area, performance and power, but also implements various techniques for exploration of the design space.

In this section, we briefly describe the parameterized VLIW platform used as testbed for the experiments, the general evaluation flow along with the high-level estimation models used to evaluate the performance indexes to be optimized and the set of applications used as benchmarks.

A. Reference Architecture

The parameterized system architecture used in this work is based on HPL-PD [17] which is a parametric processor meta-architecture designed for research in instruction-level parallelism of EPIC/VLIW architectures[1]. The HPL-PD opcode repertoire, at its core, is similar to that of a RISC-like load/store architecture, with standard integer, floating point (including fused multiply-add type operations) and memory operations.

Architectural parameters can be classified in three main categories: *register files*, *functional units* and *memory sub-system*. The first two depend on the implementation of the VLIW core and regard the size of the register files, in terms of the number of registers contained in each of them, and the number of functional units for each type of unit supported. As far as the former are concerned, five different types of register files can be identified: GPR (32-bit registers for integers), FPR (64-bit registers for floating point values) PR (1-bit registers used to store the Boolean values of predicated instructions), BTR (64-bit registers containing information about possible future branches) and CR (32-bit control registers containing information about the internal state of the processor). The functional units involved are: *Integer units*, *floating point units*, *memory units* (associated with load/store operations) and *branch units* (associated with branch operations). With respect to the memory sub-system, the parameters that can be modified are the *size*, *associativity* and *block size* for each of the three caches: First-level data cache (L1D), first-level instruction cache (L1I) and second-level unified cache (L2U).

B. Evaluation Flow

We will now show a functional scheme highlighting the main blocks of the Epic-Explorer [15] environment and the interface with the Trimaran tools. The input for the whole evaluation flow is the source of the application involved in the optimization and the configuration of the architecture being evaluated. With reference to Figure 1 this input is represented by the blocks *App.c* and *Config*.

The application (App.c) is first compiled by the Trimaran's compiler front-end (IMPACT). This front-end performs ANSI C parsing, code profiling, classical code optimizations and block formation. The intermediate code produced, together with the High Level Machine Description Facility (HMDES) machine specification [18], represents the Elcor input. The HMDES is the

[1]EPIC is an extension of the VLIW approach; where it is not necessary to make a distinction we will only use the term VLIW for the sake of simplicity.

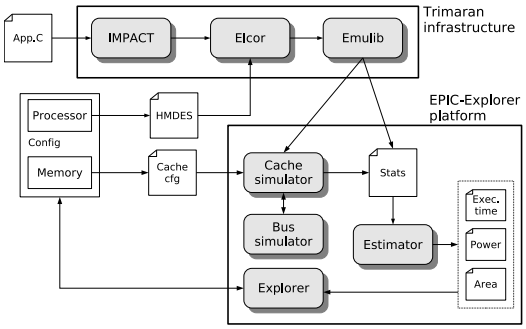


Fig. 1. Block diagram of the framework.

TABLE I
EVALUATION TIME FOR A SIMULATION (COMPILATION + EXECUTION) FOR SEVERAL MULTIMEDIA BENCHMARKS ON A OPTERON 2.6 GHZ LINUX WORKSTATION.

Benchmark	Description	Input size(KB)	Evaluation Time(sec)
wave	Audio Wavefront computation	625	7.7
g721-enc	CCITT G.711, G.721 and G.723 voice compressions	8	25.9
jpeg-codec	jpeg compression and decompression	128	33.2
mpeg2-dec	MPEG-2 video decoding	400	143.7
adpcm-enc	Adaptive Differential Pulse Code Modulation speech encoding	295	22.6
adpcm-dec	Adaptive Differential Pulse Code Modulation speech decoding	16	20.2
fir	FIR filter	64	9.1

machine description language used in Trimaran, which describes a processor architecture from the compiler's point of view. With reference to the tunable parameters outlined in the previous subsection, the hmdes file specifies the number and type of register files and functional units in the VLIW core. Elcor is Trimaran's back-end VLIW compiler for the HPL-PD architecture and it is parameterized by the hmdes machine description. It performs three tasks: Code selection and scheduling, register allocation, and machine dependent code optimizations. At the end of the compilation flow, Trimaran supplies a simulation library (Emulib) which makes it possible to execute the VLIW code produced by Elcor, generating a file (Stats) containing the execution statistics (e.g., instruction mix, execution cycles, utilization of functional units, etc.). A cache simulator, along with a bus simulator, is used to gather information about the behavior of the memory hierarchy in terms of miss rate and data/address traffic on the interconnection buses.

Together with the configuration of the system, the statistics produced by simulation contain all the information needed to apply the area, performance and power consumption estimation model implemented in the *Estimator* component of Epic Explorer. The results obtained by these models are the input for the *Explorer* component. This component executes an optimization algorithm, the aim of which is to modify the parameters of the configuration so as to minimize the three cost functions (area, execution time and energy/power consumption).

Table I reports the computation effort needed for one evaluation (i.e. simulation) of just a single system configuration for several media and digital signal processing application benchmarks.

III. INTEGRATING EPIC-EXPLORER INTO THE HPC

The first step to integrate the Epic-Explorer suite into the HPC environment. Figure 2 shows the exploration flow on HPC environment. Trimaran required some modifications to allow for multiple processes to operate on the same disk without conflicts. A tree directory structure was employed under the workspace root with the processor configurations' identifiers as intermediate nodes and the cache configurations' as leaves. This way multiple instances of Trimaran can compile codes for different architectures under different directories without interfering with each other. A bit of care had to be taken with benchmarks needing external input, because of the multi-directory structure created the initialization sequence for the benchmarks had to be modified to integrate an in-place setup of the executable. This way the simulation library of Trimaran (emulib) runs in an environment equivalent to the original one.

We wrote the JDL (job description language) for the MPI job and started launching Epic-Explorer on the cluster, but encountered some problems during the first phases of testing. The

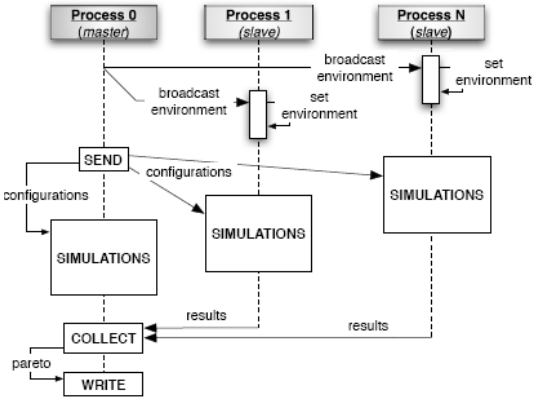


Fig. 2. Exploration flow on HPC environment.

TABLE II
DESIGN SPACE OF THE PARAMETERIZED VLIW BASED SYSTEM ARCHITECTURE.

Parameter	Parameter space
Integer Units	1,2,3,4,5,6
Float Units	1,2,3,4,5,6
Memory Units	1,2,3,4
Branch Units	1,2,3,4
GPR/FPR	16,32,64,128
PR/CR	32,64,128
BTR	8,12,16
L1D/I cache size	1KB,2KB,...,128KB
L1D/I cache block size	32B,64B,128B
L1D/I cache associativity	1,2,4
L2U cache size	32KB,64KB...,512KB
L2U cache block size	64B,128B,256B
L2U cache associativity	2,4,8,16
Space size	7.739×10^{10}

program executed correctly with 1, 2 and 4 hosts, but as soon as we required 8 hosts the program would crash reporting communication problems. Analyzing the problem we found that it was caused by an incongruence in the environment the job found on different hosts. In fact up to 4 processes can be executed on one host, and only using 8 or more we need to employ a second host. More specifically in the second host the environment variable that defined the directory in which the workspace is placed was incorrectly set by the middleware/queue manager. As we know the environment is correct on the host that starts the computation (master host), and the home directory tree gets mirrored via scp to all the hosts. So we have only to broadcast the correct environment variable from the master host to all the slave hosts, and then correct the environment before starting the actual computation.

IV. EXPERIMENTS AND RESULTS

In this section we perform an extensive evaluation of the proposed framework with an exploration of 1000 configurations of the parameterized VLIW based system architecture being studied. The configurations to be simulated were randomly chosen in the design space in Table II and executed on the cluster infrastructure described in subsection IV-A. The numerical results are presented in subsection IV-B.

A. The HPC Infrastructure

The testing was done using the following configuration: $16 \times IBMLS21$ Blades with 2 Opteron 2.6 GHz dual core processors (for a total of 4 cores per blade) equipped with 8GB of DDR and a 73GB SATA HD, linked by Gigabit Ethernet. The cluster used for the tests was configured to allow 1 process per core (4 processes per host) and one of the hosts is reserved to run cluster services and to manage the jobs, so the maximum reachable number of MPI processes was $4 \times 15 = 60$.

The software package was installed on the dedicated host that was also the cluster coordinator. All the hosts share a single distributed file system (GPFS). This eliminates the need to copy the package on all the hosts, but poses a new problem because of the I/O bottleneck in writing concurrently on a single file system.

The tests were done with no interference from other jobs in the cluster, so the scheduler tried to allocate the processes as close as possible in order to fill one host before employing the next one.

One disadvantage is that big jobs that produce a lot of output cannot be run on a single host, for example the 'mpeg2dec' benchmark produces about 130GB of output with a random exploration of 1000 configurations, this does not fit in a single HD so we had to scale down to

100 configurations to evaluate the performances for 1, 2 and 4 processes (that run on a single host). The values obtained were therefore multiplied by 10 as an approximation with linear interpolation.

B. Experiments

In Figure 3 it is plotted the total time for an exploration that requires the simulation of 1000 configurations using a growing number of processors. The total time is normalized by the time needed on a single processor. As we can see in Figure 3 even using an exponentially growing number of processors the wall clock time was reduced only by an order of magnitude.

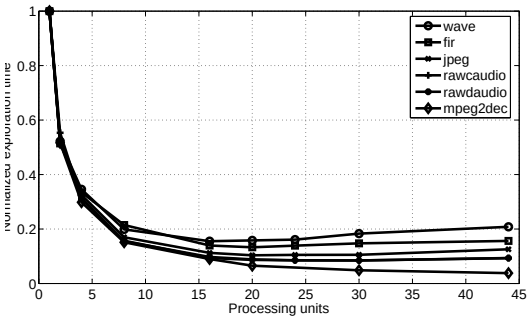


Fig. 3. Normalized time required for an exploration of 1000 configurations with an exponentially growing number of processors.

Increasing the number of processes quickly leads to the saturation of the computational times. We have a very steep performance increment passing from 1 to 2 processes, almost cutting in half the exploration time. We still get good benefits from the parallelization employing 4 or 8 processes, but soon after that we reach the saturation point. The simpler benchmarks as wave even start to get worse performances while the more complex ones like mpeg2dec still have some little benefits using a number of processes as high as 60.

An interesting point can be made looking at Figure 5, in which the efficiency of the explorations with an exponentially growing number of processors is shown. The efficiency is reported on the Y axis, expressed as the ratio of

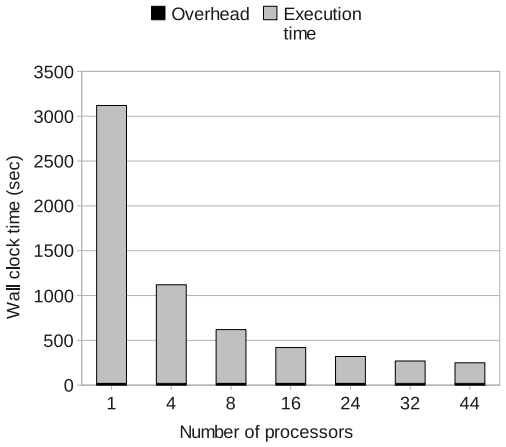


Fig. 4. Time needed to run 1000 simulations of jpeg-codec with a growing number of hosts.

cpu time to the whole exploration time used, obtained with the formula:

$$\frac{CPU_{TIME}}{(WALL_CLOCK_TIME \times N_{PROCESSES})} \quad (1)$$

Equation 1 gives us an idea of how much of the whole computing time was spent doing something useful, and indirectly an index of the efficiency of the calculation. This is also an indirect estimate of the I/O boundness of the job configuration. As we can notice we get high efficiency for 1 or 2 processes, but start to get worse results with 4 processes. After that we get again good results with 8 processors and then start to loose efficiency almost linearly. This can be interpreted as an I/O bottleneck effect, in fact up to 4 processes we are adding more CPU power but we are using the same disk, so we quickly fill the available disk bandwidth resulting in more idle wait. As soon as we get to 8 processes the benchmarks' efficiency grows again because of the added disk, while simpler benchmarks as wave that do not produce a lot of output are less affected by this.

V. Conclusions

In this paper we present a case study of design space exploration of a complex highly parameterized VLIW based SoC platform. We

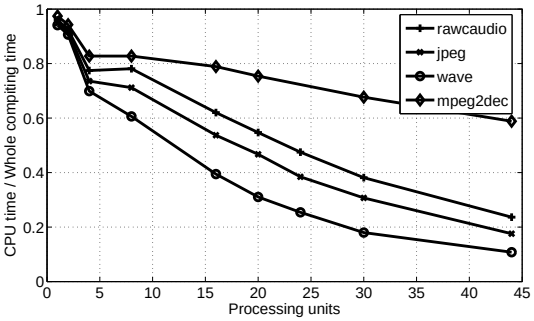


Fig. 5. Efficiency ratio for an exploration of 1000 configurations with an exponentially growing number of processors.

test the use of High Performance Computing (HPC) as a viable solution to tackle with DSE related problems. A reduction of up to one order of magnitude in exploration time has been observed in our experiments. Obtained results remark that the use of HPC environments is likely to be one of the most promising way to significantly reduce the exploration time. Future works may involve the exploration of different ways of employing the parallelism offered by HPC/Grid environments through finely tuned parallel algorithms (i.e. pMOEA).

REFERENCES

[1] K. Keutzer, S. Malik, R. Newton, J. M. Rabaey, and A. Sangiovanni-Vincentelli, "System-level design: Orthogonalization of concerns and platform-based design," *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, vol. 19, no. 12, pp. 1523–1543, Dec. 2000.

[2] F. Vahid and T. Givargis, "Platform tuning for embedded systems design," *IEEE Computer*, vol. 34, no. 3, pp. 112–114, Mar. 2001.

[3] P. Nsame and Y. Savaria, "A customizable embedded soc platform architecture," in *4th IEEE International Workshop on System-on-Chip for Real-Time Applications*, pp. 299–304.

[4] T. Givargis, F. Vahid, and J. Henkel, "System-level exploration for Pareto-optimal configurations in parameterized System-on-a-Chip," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 10, no. 2, pp. 416–422, Aug. 2002.

[5] W. Fornaciari, D. Sciuto, C. Silvano, and V. Zaccaria, "A sensitivity-based design space exploration methodology for embedded systems," *Design Automation for Embedded Systems*, vol. 7, pp. 7–33, 2002.

[6] G. Ascia, V. Catania, and M. Palesi, "A multi-objective genetic approach for system-level exploration in parameterized systems-on-a-chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 4, pp. 635–645, Apr. 2005.

[7] G. Hekstra, D. L. Hei, P. Bingley, and F. Sijstermans, "TriMedia CPU64 design space exploration," in *International Conference on Computer Design*, Austin Texas, Oct. 10–13 1999, pp. 599–606.

[8] S. G. Abraham, B. R. Rau, and R. Schreiber, "Fast design space exploration through validity and quality filtering of subsystem designs," HP Laboratories Palo Alto, Tech. Rep. HPL-2000-98, July 2000.

[9] R. Szymanek, F. Catthoor, and K. Kuchcinski, "Time-energy design space exploration for multi-layer memory architectures," in *Design, Automation and Test in Europe*, 2004, pp. 181–190.

[10] S. Neema, J. Sztipanovits, and G. Karsai, "Design-space construction and exploration in platform-based design," Institute for Software Integrated Systems Vanderbilt University Nashville Tennessee 37235, Tech. Rep. ISIS-02-301, June 2002.

[11] A. Ghosh and T. Givargis, "Cache optimization for embedded processor cores: An analytical approach," *ACM Transactions on Design Automation of Electronic Systems,*, vol. 9, no. 4, pp. 419–440, Oct. 2004.

[12] L. Eeckhout, S. Nussbaum, J. E. Smith, and K. D. Bosschere, "Statistical simulation: Adding efficiency to the computer designer's toolbox," *IEEE Micro*, vol. 23, no. 5, pp. 26–38, Sept.-Oct. 2003.

[13] S. Eyerman, L. Eeckhout, and K. D. Bosschere, "Efficient design space exploration of high performance embedded out-of-order processors," in *DATE '06: Proceedings of the conference on Design, automation and test in Europe*. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2006, pp. 351–356.

[14] G. Ascia, V. Catania, A. G. D. Nuovo, M. Palesi, and D. Patti, "Efficient design space exploration for application specific systems-on-a-chip," *Journal of Systems Architecture*, vol. 53, pp. 733–750, Oct. 2007.

[15] G. Ascia, V. Catania, M. Palesi, and D. Patti, "EPIC-Explorer: A parameterized VLIW-based platform framework for design space exploration," in *First Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia)*, Newport Beach, California, USA, Oct. 3–4 2003, pp. 65–72.

[16] "An infrastructure for research in instruction-level parallelism," http://www.trimaran.org/.

[17] V. Kathail, M. S. Schlansker, and B. R. Rau, "HPL-PD architecture specification: Version 1.0," Compiler and Architecture Research HP Laboratories Palo Alto HPL-93-80, Tech. Rep., 2000.

[18] J. Gyllenhaal, "A machine description language for compilation," Master's thesis, Department of Electrical and Computer Engineering, University of Illinois, Urbana IL, Sept. 1994.

A Non-Parametric Parallel Harris-Affine Detector

F. Bellavia[1,2], D. Tegolo[1,2], C. Valenti[1,2]
[1]*Consorzio COMETA, Via Santa Sofia 64, 95123 Catania, Italy*
[2]*Università di Palermo, Via Archirafi 34, 90128 Palermo, Italy*
{fbellavia, domenico.tegolo, valenti}@unipa.it

***Abstract*—This paper describes a parallel version of a new automatic Harris-based corner detector. A simple but effective client-server based scheduler has been implemented in order to dynamically distribute the workload on heterogeneous parallel architectures such as Grid systems. Results obtained on the COMETA Grid show the effectiveness and the robustness of the proposed approach.**

***Index Terms*—COMETA, parallel scheduling, image feature detectors.**

I. Introduction

MANY image analysis applications as three-dimensional reconstruction [1], mosaicing [2], object recognition and classification [3] rely on feature detection as a fundamental step. In this paper a parallel version of an automatic Harris-based feature detector which has been proved to gives better results compared to the standard Harris-affine detector, but with higher computational cost, has been implemented. To avoid the bottlenecks due to the characteristic input data, a new parallel scheduler has als been proposed. It balances and dynamically distributes the computation, trying to guess the next inactivity of the processors and sets new rendezvouses to collect statistical information to choose the best balancing scheme.

An introduction of the feature detector is described in Sec. II, while the parallel scheduler and details about the parallel implementation are reported in Sec. III. Conclusions and final remarks can be found in Sec. IV.

II. Non-parametric Harris-based Detector

The Harris detector [4] is one of the first detectors used in digital image analysis. Although it is only rotational invariant, in the last years new Harris-based algorithms have been developed which are scale and affine invariant, like the Harris-Laplace [5] and Harris-affine [5], [6] detectors.

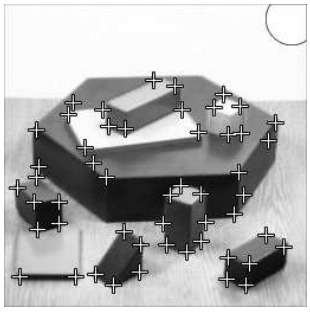


Fig. 1. Harris corners superimposed on a image.

These algorithms can be used to find characteristic regions in similiar images [7], allowing automatic or semi-automatic tasks like objects detection and classification [3], or three-dimensional recostruction [1], [8] and mosacing [1], [2].

An Harris corner (see Fi. 1) corresponds to a region of an input image I with locally maximized directional derivatives, within a neighborhood N, in any orthogonal basis (i.e. I_x and I_y). The Harris function

$$H(\mathbf{p}) = \det\big(\mu(\mathbf{p})\big) - \alpha\ \mathrm{trace}^2\big(\mu(\mathbf{p})\big)$$

where

$$\mu(\mathbf{p}) = \begin{bmatrix} \sum\limits_N I_x^2 & \sum\limits_N I_{xy} \\ \sum\limits_N I_{xy} & \sum\limits_N I_y^2 \end{bmatrix}$$

returns higher values in the case of a corner $\mathbf{p}$, since the determinant stresses both edges and corners, while the trace puts in evidence edges only. The α parameter and a successive threshold value t must be manually set to avoid false corner points. It must be noted that both α and t are very dependent on the input image.

In the *affine scale-space theory* [9] matrix μ becomes

$$\mu(\mathbf{p},\sigma_\mathrm{D},\sigma_\mathrm{I}) = \sigma_\mathrm{D}^2\, g(\cdot,\sigma_\mathrm{I}) \otimes \begin{bmatrix} \sum\limits_N L_x(\mathbf{p},\sigma_\mathrm{D})L_x(\mathbf{p},\sigma_\mathrm{D}) & \sum\limits_N L_x(\mathbf{p},\sigma_\mathrm{D})L_y(\mathbf{p},\sigma_\mathrm{D}) \\ \sum\limits_N L_x(\mathbf{p},\sigma_\mathrm{D})L_y(\mathbf{p},\sigma_\mathrm{D}) & \sum\limits_N L_y(\mathbf{p},\sigma_\mathrm{D})L_y(\mathbf{p},\sigma_\mathrm{D}) \end{bmatrix}$$

where σ_D and σ_I are the *differential scale* and the *integration scale*, respectively. The operator $\otimes$ represents the convolution between a Gaussian kernel with standard deviation σ_I. $L_x(\mathbf{p},\sigma_\mathrm{D})$ (or $L_y(\mathbf{p},\sigma_\mathrm{D})$) is the derivative on $\mathbf{p}$ along the direction x (or y) after another Gaussian convolution with standard deviation σ_D.

A new version of the Harris corner detector has been introduced in [11] to avoid the use of the parameters α and t. A local edge enhancement, namely *weighted z-score* [10], is computed by weighting $L_x(\mathbf{p},\sigma_\mathrm{D})$ on a Gaussian window $W(L_x(\mathbf{p},\sigma_\mathrm{D}),\sigma_\mathrm{I})$, centered in $\mathbf{p}$ and with radius σ_I

$$Z_x(\mathbf{p},\sigma_\mathrm{D},\sigma_\mathrm{I}) = \frac{L_x(\mathbf{p},\sigma_\mathrm{D}) - \overline{W}(L_x(\mathbf{p},\sigma_\mathrm{D}),\sigma_\mathrm{I})}{\mathrm{std}(W(L_x(\mathbf{p},\sigma_\mathrm{D}),\sigma_\mathrm{I}))}$$

where $\overline{W}$ and $\mathrm{std}(W)$ are the mean and standard deviation of the weighted intensities of the gradient in W.

Let us indicate with $G(\mathbf{p})$ and $G_z(\mathbf{p})$ the normalized versions, in range $[0,1]$, of the gradient magnitude

$$G(\mathbf{p},\sigma_\mathrm{D}) = \sqrt{L_x^2(\mathbf{p},\sigma_\mathrm{D}) + L_y^2(\mathbf{p},\sigma_\mathrm{D})}$$

and of the z-scored gradient

$$G_z(\mathbf{p},\sigma_\mathrm{D},\sigma_\mathrm{I}) = \sqrt{Z_x^2(\mathbf{p},\sigma_\mathrm{D},\sigma_\mathrm{I}) + Z_y^2(\mathbf{p},\sigma_\mathrm{D},\sigma_\mathrm{I})}.$$

These two new images are multiplied pixel by pixel to weight each other

$$P(\mathbf{p}) = G(\mathbf{p})G_z(\mathbf{p})$$

and further processed by a median filter with radius $3\sigma_\mathrm{D}$, in order to enhance the contrast, obtaining $P'(\mathbf{p})$. The mask M

$$M(\mathbf{p}) = g(\cdot,\sigma_\mathrm{D}) \otimes B(\mathbf{p})$$

where

$$B(\mathbf{p}) = \begin{cases} 0 & P'(\mathbf{p}) \leqslant \overline{P'} \\ 1 & P'(\mathbf{p}) > \overline{P'} \end{cases}$$

identifies the edges discards false corners.

The new Harris function is obtained by replacing L_x and L_y in μ with:

$$M_x(\mathbf{p},\sigma_\mathrm{D}) = M(\mathbf{p},\sigma_\mathrm{D})Z_x(\mathbf{p},\sigma_\mathrm{D})$$

and

$$M_y(\mathbf{p},\sigma_\mathrm{D}) = M(\mathbf{p},\sigma_\mathrm{D})Z_y(\mathbf{p},\sigma_\mathrm{D}).$$

Next we compute the z-score function on the whole matrix to avoid the introduction of α:

$$H_z(\mathbf{p}) = Z(\det(\mu(\mathbf{p}))) - Z(\mathrm{trace}^2(\mu(\mathbf{p}))$$

where

$$Z(K(\mathbf{p})) = \frac{K(\mathbf{p}) - \overline{K}}{\mathrm{std}(K)}$$

is defined here for a generic image K. After this step the determinant and the squared trace are comparable. Finally, a pixel $\mathbf{p}$ is selected as a corner if:

- $H_z(\mathbf{p}) > 0$;

- $H_z(\mathbf{p})$ is a local maximum for H_z;
- $M(\mathbf{p}) > 0.31$, to avoid the effect of point-like noise (this threshold can be obtained theoretically and it does not depend on σ_D).

The proposed method gives better results compared to the standard Harris-affine detector and does not introduce any user-given threshold value, although more computational time is required [11].

In implementing the parallel version of the detector, an unbalanced workload distribution could afflict a trivial implementation. Indeed, time required to traverse the data structure, used in the statistical filters (i.e. median and max filters), strictly depends on the data which usually varies among the regions of the image. Therefore a scheduler which distributes data on runtime has been implemented.

III. Parallel Implementation

A parallelization of the algorithms has been done in order to take advantage of the highest performance. We tried to optimally distribute the workload on all free processors that become available during the computation itself.

We have adopted the strategy of separated I/O and CPU bound tasks, which can run concurrently, to speedup the system. *Communication* and *working threads* are the only threads that run: the former takes care of the messages among processors, while the latter elaborates the data, according to the information given by its own communication thread. Communication threads have been defined by multithreaded server-client MPI [12], [13] routines. The distribution policy is, based on statistics obtained on run time. The pseudo codes of these threads are shown in the listings of Figs. 2–4.

Communication and working threads share a common memory to pass the data. This leads to critical regions when the communication thread asks or transmits to its worker about the amount of remaining job.

The scheduler we have defined to manage the processors workload is simple, but effective. First of all, it distributes the workload to all

```
server_communication thread
do
  lock mutex;
  query its own working thread about the remaining job;
  unlock mutex;
  query all clients the amount of remaining job;
  if the server working thread finished its job then
    send the output to the relevant working thread;
  if there is no data to process then
     wake the server working thread up through mutex;
     exit;
  end;
  schedule and send the jobs to the clients;
  lock mutex;
  notify the remaining job to its own working thread;
  unlock mutex;
  send required data to the free working threads;
  if the server working thread has data to work with then
    set the working thread to busy;
    wake the working thread of the server up through mutex;
  lie idle until next scheduled time;
while true.
```

Fig. 2. Pseudo code of the server communication thread.

```
client_communication thread
do
  lock mutex;
  query its own working thread sbout the remaining job;
  unlock mutex;
  send the amount of remaining job to the server;
  if its own working thread has finished its job then
    send the output to the relevant working thread;
  if there is no data to process then
     wake its own working thread up through mutex;
     exit;
  end;
  get the job from the server;
  lock mutex;
  notify the remaining job to its own working thread;
  unlock mutex;
  send required data to the free working threads;
  if its own working thread has data to work with then
    set its own working thread to busy;
    wake its own working thread up through mutex;
  wait the server for a request of data;
while true.
```

Fig. 3. Pseudo code of the client communication thread.

```
worker thread
do
  set current status to free;
  run its own server/client;
  wait by mutex until the status is not free;
  while the job has to be completed
    process a single data;
    lock mutex;
    update the advancement of the job;
    unlock mutex;
  end;
  if there is no data to process then
    exit;
while true.
```

Fig. 4. Pseudo code of the worker thread.

free processors. Next, it estimates the average time, by using an *exponentially weighted moving average* [14], and assigns online half of the workload of the processor that should end last to the remaining free processors. This approach is repeated until there are no more free processors (see Fig. 6). Successive management is done after half of the expected time of the current processor that should end first. If no progress is obtained, then the elapsed time is added to the expected average time. An efficient implementation of this procedure has been obtained by a heap structure.

As for the communication and working threads, we report the pseudo code relative to the whole algorithm (see Fig. 5).

```
   procedure parallel_harris
 1 assign a scale to each process;
 2 for each scale
 3   compute Z_x, G, G_z, P;
 4 collect P from all processes;
 5 apply the server/client and working threads:
 6   compute the median filter P' on all images P;
 7 for each scale
 8   compute B, M, H, H_z;
 9 collect H_z from all processes;
10 apply the server/client and working threads:
11   compute the max filter on all images H_z;
12 for each scale
13   locate the corner candidates;
14 collect the final corners from all processes.
```

Fig. 5. Pseudo code of the worker thread.

To locate the initial corners, a scale is assigned to each running processor separately, but the statistical filters are computed simultaneously by all processors.

The starting phase of the algorithm described makes use of statistical filters on floating point values with a variety of kernel size, which depends on the scale. The performance dramatically lessens when the size of the scale increases. In order to compute the statistical filters, the image is split into overlapping horizontal stripes with different heights and assigned to all processors, according to their performances. A heap is used to compute the max filter, while a 3-way quick-select method is used for the median filter, instead of the approximate algorithm proposed in [15].

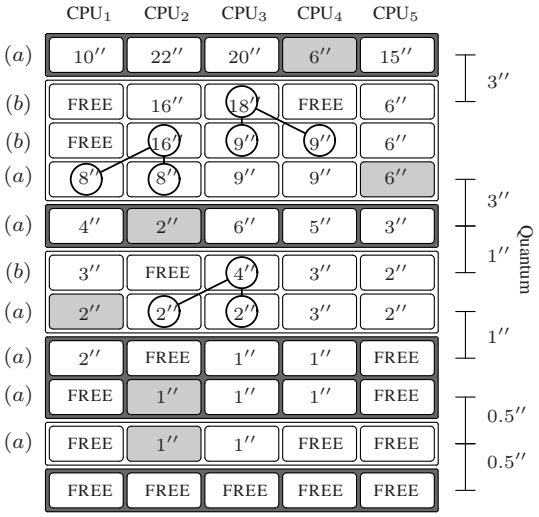


Fig. 6. Last seven sample atomic actions of the scheduler on five processors are represented (from top to down). *a*: When no free processor is available, then the processor that should end first is highlighted in gray: half of its expected running time is used as the following quantum by the scheduler. *b*: If a processor is free, then it receives half of the workload of the processor that should end last and that should spend more than the quantum.

IV. Discussion and Results

A parallel version of an automatic algorithm for corner detection has been presented in this paper. An ad hoc scheduler dynamically balances the workload on the working threads of free processors.

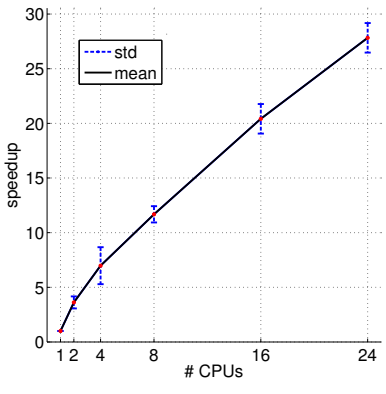


Fig. 7. Parallel implementation speedup.

At present, this scheduler assumes that all

processors are equal, but a new version of this workload manager will consider also heterogeneous processors on Grid systems. In figures 7–8 the average speedup and the running time are shown.

These results have been obtained on 8 different images by using the COMETA Grid and report standard deviations in the form of vertical bars and highlight the good performances of the parallel implementation.

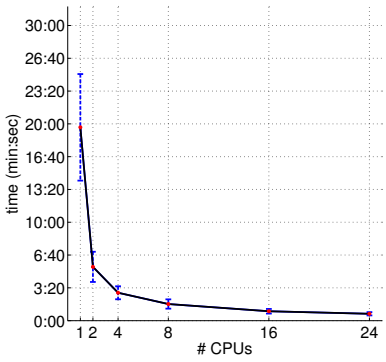


Fig. 8. Parallel implementation computational time.

To effectively compare the performances of the new proposed scheduler, a static parallel version of the new parallel corner detector, has been implemented, too. In the static implementation, equal horizontal stripes are distributed at the beginning of each statistical filter computation. Fig. 9 shows that the dynamic scheduler performs better, except for the case of 2 processors, due to communication overhead.

REFERENCES

[1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, 2004.

[2] M. Brown and D.G. Lowe, *Recognising Panoramas*, International Conference on Computer Vision, 2003.

[3] K. Mikolajczyk, B. Leibe and B. Schiele, *Local Features for Object Class Recognition*, International Conference on Computer Vision, 2005.

[4] C.G. Harris and M. Stephens, *A Combined Corner and Edge Detector*, 4th Alvey Vision Conference, 1988.

[5] K. Mikolajczyk and C. Schmid, *Scale and Affine Invariant Interest Point Detectors*, International Journal of Computer Vision, 2004.

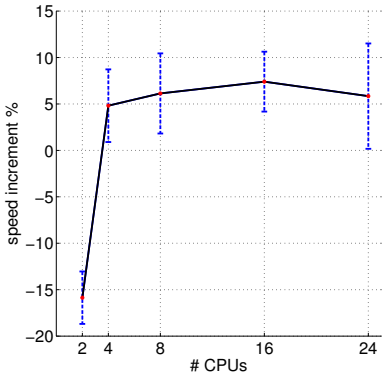


Fig. 9. Performances increment of the proposed scheduler over a static one.

[6] A. Baumberg, *Reliable Feature Matching Across Widely Separated Views*, Conference on Computer Vision and Pattern Recognition, 2000.

[7] K. Mikolajczyk and C. Schmid, *A Performance Evaluation of Local Descriptors*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

[8] F. Schaffalitzky and A. Zisserman, *Multi-view Matching for Unordered Image Sets, or How Do I Organize My Holiday Snaps?* , European Conference on Computer Vision, 2002.

[9] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Kluwer Academic Publishers, 1994.

[10] M.H. DeGroot and M.J. Schervish, *Probability and Statistics*, 3rd ed., Addison Wesley, 2001.

[11] F. Bellavia, D. Tegolo and C. Valenti, *A non-parametric scale-based corner detector*, International Conference on Pattern Recognition, 2008.

[12] W. Gropp, E. Lusk and A. Skjellum, *Using MPI: Portable Programming with the Message-Passing Interface*, 2nd ed., MIT Press, 1999.

[13] W. Gropp et al., *MPI - The Complete Reference. Volume 2: The MPI-2 Extensions*, MIT Press, 1998.

[14] J.S. Hunter, *The exponentially weighted moving average*, Journal of Quality Technology, 1986.

[15] B. Wilkinson and M. Allen, *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*, 2nd ed., Prentice Hall, 2004.

F. Bellavia is a Computer Science Ph.D. student at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include image analysis and, in particular, three-dimensional reconstruction and parallel programming.

D. Tegolo is an Associate Professor at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include image analysis, focusing on medical images, artificial vision and parallel architectures.

C. Valenti is an Assistant Professor at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include image analysis and, in particular, tomographic reconstruction, evolutionary programming and parallel systems.

A Parallel Automatic Three-dimensional Fuzzy Edge Detector

M. Cipolla[1,2], V. Di Gesù[1,2], C. Valenti[1,2], B. Zavidovique[3]
[1]*Consorzio COMETA, Via Santa Sofia 64, 95123 Catania, Italy*
[2]*Università di Palermo, Via Archirafi 34, 90128 Palermo, Italy*
{mcipolla, vitodigesu, valenti}@unipa.it
[3] *Institut d'Electronique Fondamentale, Universitè Paris XI, France,*
bertrand.zavidovique@u-psud.fr

***Abstract*—We investigate the segmentation of three-dimensional objects represented by manifold meshes. Our goal is to locate, on the surface of the object, regions characterized by high flexure, usually named edges. The proposed method is automatic and the analysis of the object is performed by using a proper scanning window, centered on each mesh element. The algorithm returns representative edge-type structures of the object, and can be used as a pre-processing step for further application, such as 3D compact representation, matching and recognition. A parallel implementation to analyze very large mesh will be described.**

***Index Terms*—Mesh segmentation, parallel algorithms. Cometa**

I. INTRODUCTION

Three-dimensional data can be generated by using different acquisition modalities, such as laser scanners and multiple view, and different data representation are required depending on the nature of the acquired object. We deal here with triangular meshes, defined as $M = \{V, E, F\}$, where V, E, and F represent the sets of *vertices*, *edges* and *facets* of the model surface, respectively. The proposed edge detector is based on a fuzzy approach. The process infers a natural segmentation of the surface process by defining fuzzy sets based on the so called *saliency* of each arc on the surface, the saliency allows to define the fuzzy membership function on a continuous domain. Section II briefly reviews the state of the art algorithms for edge detection and illustrates our methodology. Section IV reports conclusions on the experimental results and introduces possible future works. Section III shows our approach for the parallelization of the algorithm.

II. 3D EDGE DETECTION

Three-dimensional segmentation is usually used for locating representative structures on the surfaces of the objects. Literature proposes different methodologies, unfortunately, these methods require thresholding and are not fully automatic. Two watershed approaches are proposed in [1], [2]. Mean curvature flow and anisotropic diffusion are used in [4], where the authors give a theoretic formulation for fairing and segmenting 3D surfaces. Nevertheless, this method, in order to perform the segmentation stage, needs the tuning of a threshold parameter.In this paper we propose an heuristic approach, our algorithm is able to extract salient features from the surface of an object by performing a surface analysis based on a weighting function which computes a measure of the local surface bending near each element of the mesh. Different weighting functions, namely *Second Order Difference Operator*, *Extended Second Order Difference Operator* [3] and *tensor and normal voting* [2], have been defined, they require different user given parameters that depend on the mesh to analyze. Moreover, real

objects can be acquired at different samplings by laser scanner devices and the resulting mesh can be modified by using simplification techniques [5].The parameter tuning process is time-consuming, subjective and error-prone, thus an automatic methodology, able to return stable three-dimensional segmentations, is highly desirable.

For an arc $a \in E$, let $L_i(a) = \{a' : \text{such that the length of the minimum path from } a \text{ to } a' \text{ is } i\}$, that is $L_0(a) = a$ and $L_0(a)$, $L_1(a)$, $L_2(a)$,... are disjointed layers. We define the neighborhood $N_r(a)$ with radius r of a as $N_r(a) \triangleq \bigcup_{i=0}^{r} L_i(a)$. The area covered by $N_r(a)$ (i.e. the sum of the areas of the triangles in $N_r(a)$) will be indicated by $A_r(a)$.

According to the second order difference operator, the weight $w(a)$ of the arc a, shared by two facets f_1 and f_2, with normal versors n_1 and n_2 respectively, is equal to the dihedral angle between them:

$$w(a) = \arccos(n_1 \cdot n_2)$$

where $(\cdot)$ indicates the usual inner product. In the case of the extended second order difference operator, the normal versors n_1 and n_2 are computed as average normal vectors to the facets that share the vertices, $v_1 \in f_1$ and $v_2 \in f_2$, opposite to a.

Our method classifies a by analyzing the distribution of w, with respect to the neighborhoods $N_r(a)$, of increasing radius r: an edge separates different smooth surfaces which, on average, show a decreasing trend for w. Arcs close to edges, namely *ramps*, have instead an increasing trend for w. Smooth surfaces, which do not significantly bend, are characterized by a constant trend of w.

The saliency is computed by taking into account the variance of the weights falling into the window $N_r(a)$, centered on an arc on a. Let denote with $\nu_r(a)$ the variance of the weights w, the saliency $s(a)$ is defined as:

$$s(a) = \sum_{i>0} \phi_i(a) \quad \text{with}$$

$$\phi_i(a) = \eta_i(a) \frac{\partial \nu_i(a)}{\partial r} |\nu_1(a) - \nu_i(a)|$$

where $\eta_i(a) = 1 \setminus e^{A_{\min} - A_i(a)}$ is used to control the resulting values ϕ_i, when moving far from a, with respect to the smallest window of radius $r = 1$. Note that $A_{\min} = \min_{a \in E}\{A_1(a)\}$.

The effect of the normalization coefficient η, is to avoid considering structures too far from the center a of the neighborhood and ensures a coherent saliency even in the case of meshes with variable samplings which results in facets with different dimensions. Though the whole mesh should be scanned to compute s, it can be seen that, due to the effect of η the contribution of large radii is close to zero and can be neglected.

The derivative $\frac{\partial \nu_i(a)}{\partial r} = \nu_i(a) - \nu_{i-1}(a)$ measures the trend of change of the variance, sharp edges are characterized by a fast decreasing f and, vice versa, smooth details have slower decreasing f. The term $|\nu_1(a) - \nu_i(a)|$ is needed because it is possible for two given f trends to have locally similar derivatives, though different values of f should suggest different values of s. Actually, negative values of s correspond to edges, positive values indicate ramps, while values close to zero denote smooth surfaces. Therefore, the saliency formula s assigns a score to each arc a, thus to discriminate among edges, ramps and smooth surfaces. In order to classify the arcs into these three categories, we have implemented a fuzzy set approach. The fuzzy sets $(S_{\mathrm{E}}, m_{\mathrm{E}})$,$(S_{\mathrm{R}}, m_{\mathrm{R}})$ and $(S_{\mathrm{S}}, m_{\mathrm{S}})$ identify edges, ramps and smooth surfaces, respectively. The $[0, 1]$-valued membership $m(a)$ of a given arc a was defined as the maximum value of its memberships $m_{\mathrm{E}}(a)$, $m_{\mathrm{R}}(a)$ and $m_{\mathrm{S}}(a)$. For the sake of clarity, we indicate edges in blue, ramps in red and smooth zones in green.

III. Parallel approach

We derived great benefit of the GRID system by the Cometa Consortium. This infrastructure allowed us a fast testing of the algorithm on different data-sets of three-dimensional meshes. In particular, we developed a parallel implementation as a test-bed for a general frame-

work for mesh segmentation, based on elements neighborhood analysis. The implementation of a parallel algorithm on triangle mesh is a complex task. The main issue is about the partition algorithm to use and the management of the boundary elements. Different solution have been developed [6], [7], meshes are usually divided off-line and the partition are distributed among the processors. Although partitions can be created with some overlap, the dimension of the neighborhood of an element cannot be known a priori. Our approach is based on a divide-merge strategy, the mesh is first divided by a *recursive binary division* and then analyzed, skipping boundary elements that are lately processed during the merge steps. The division method creates a binary tree pattern of communication between processors, we use this hierarchic structure for distribute the required division steps among the processors. In order to process a whole data-set of meshes we had to exploit a cluster of processors through its structure: to reduce the communication overhead and to distribute the workload to each node, according to its performance. Nodes in a cluster are organized in sub-groups, where the elements of each group share resources. This led to inter-communication and intra-communication: the former is performed between nodes belonging to different sub-clusters, while the latter involves nodes on the same sub-cluster. The communication rate is higher for intra-communications, thus is desirable to balance the workload to reduce inter-communications.

We suggest *training-based* approach to capture the structure of the Grid. This method is general enough as it can be used for different configurations. The training is performed by using a mesh sample as test object to estimate the execution time of each processor and the transmission time between each pair of nodes. Let us consider the cluster as a complete weighted graph $G(V, E, w)$, where the weight of each arc is given by the average transmission time between its end-points. The detection of sub-clusters is performed by an agglomerative scheme: we set each sub-cluster to contain one node, then use the costs to built larger sub-clusters. A node is inserted into a sub-cluster if the average transmission time between the node itself and the nodes in the sub-cluster is not greater (up to a tolerance factor) than the average transmission time between the nodes in the sub-cluster. This is performed by estimating the transmission time and then by sorting the adjacency list of each node according to the cost of the relative arc.

An approximate division time for each data load can be achieved by interpolating the data collected from the test mesh division. The execution time for extracting neighborhoods with predefined radii, for each element of the test mesh, lets us compute the cost of a sub-cluster together with the communication and division time. The communication pattern for the nodes belonging to a sub-cluster is modeled by a binary tree. The partition process starts from the root and the chunks of data are transmitted to child nodes recursively. The division steps are thus distributed among processors. Note that if the number of nodes in a sub-cluster is m, the height of the binary tree is at most $\lfloor \log_2 m \rfloor + 1$. Let denote with T a sub-cluster with eight h_T, let $C_T(N)$ be the average communication time of the nodes belonging to sub-cluster T, when transmitting a data block of size N (namely the size of a mesh chunk, in bytes), let $D_T(n)$ the division estimated time required to split n arcs, finally let $K_T(n)$ the estimated time required to extract the neighborhoods of n arcs, for a variety of radius values. We allow a division step for the sub-cluster T if the inequality

$$D_T(n) + C_T(N) < K_T(n)$$

is satisfied. Finally, for a mesh of size N with n arcs, the time spent by a sub-cluster T to distribute and process the data is about:

$$K_T(n/2^{h_T - t}) + \sum_{i=1}^{h_T} D_T(n/2^{i-1}) + C_T(N/2^i)$$

where $t = 0$ if the number of nodes in T is a power of 2, $t = 1$ otherwise. Sub-clusters are themselves considered as nodes of a binary tree, and the same pattern of communication is

achieved by balancing the tree according their cost. Such analysis is done just once, unless the cluster configuration changes.

IV. Experimental results

Figure and 1 report a qualitative comparison between objects segmented through the second order difference operator and its extended version, which are parameter-free. The use of the former has to be preferred because it detects small, but usually important details. Indeed, this means also that this function is more sensible to noise, while the latter tends to erase small components because it performs an averaging on the normal vectors. The algorithm is resolution independent and automatic, since it accepts just the mesh to analyze and returns a reasonable segmentation without predefined or user-tuned parameters. Figure 2 shows the same object at different resolutions, obtained by reducing the original number of vertices down to 25%. Obviously, many details are missing on this rough surface, but nonetheless our algorithm is almost able to detect the same most evident features, as for any other resolution.

Ad hoc aggregation procedures group the output into distinctive features. In our case, to isolate the three-dimensional edges, we introduced a fuzzy measure, based on the saliency degree. This approach takes advantage in locating in an intuitive fashion important zones on the surface of the objects (e.g. its edges and ramps). Lastly, we will test also the goodness of set S_{E}, obtained by our method, as a subsample of the object's surface to solve problems as object alignment and symmetry detection.

V. Conclusion

Future work will focus on the definition of geometric descriptors of the features, for recognition and matching tasks. Lastly, we will test alsothe goodness of set S_{E}, obtained by our method, as a sub-sample of the object's surface, to solve object alignment and symmetry detection problems.

References

[1] L. Chen and N.D. Georganas.*An efficient and robust algorithm for 3d mesh segmentation*. Multimedia Tools and Applications, 29(2):109125, 2006.

[2] Y. Sun, D.L. Page, J.K. Paik, A. Koschan and M.A. Abidi. *Triangle Mesh-Based Edge Detection and its Application to Surface Segmentation and Adaptive Surface Smoothing*. In Proceedings of the International Conference on Image Processing, volume III, pages 825 828, 2002.

[3] A. Hubeli, K. Meyer, and M. Gross. *Multiresolution Feature Extraction for Unstructured Meshes*. In Conference on Visualization, pages 287294, 2001.

[4] U. Clarenz, G. Dziuk and M. Rumpf. *On generalized mean curvature flow in surface processing*. Geometric Analysis and Nonlinear Partial Differential Equations, pages 217248, 2003.

[5] M. Garland and P. S Heckbert. *Surface simplification using quadric error metrics*. International Conference on Computer Graphics and Interactive Techniques, pages 209216, 1997.

[6] J. Chen, V.E. Taylor, *ParaPART: A Partitioning Tool for Distributed Systems*, Proceedings of the 11 IPPS/SPDP'99 Workshop in Conjuction with the 13th International Parallel Processing Symposium and 10th Symposium on Parallel Distributed Processing, pp. 996-1005, 1999.

[7] D. Brosky, J.B. Pedersen, *A Parallel Framework for Simplification of Massive Meshes*, Proceedings of the 2003 IEEE Symposium an Parallel and Large-Data Visualization and Graphics, pp. 4, October 2003.

M. Cipolla is a Computer Science Ph.D. student at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include parallel programming and three-dimensional analysis, reconstruction and segmentation.

C. Valenti is an Assistant Professor at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include image analysis and, in particular, tomographic reconstruction, evolutionary programming and parallel systems.

V. Di Gesù is a Full Professor at the Department of Mathematics and Applications of the University of Palermo (Italy). His main research interests include image analysis, fuzzy sets, mathematical morphology and evolutionary programming.

B. Zavidovique is a Full Professor at the Institut d'Electronique Fondamentale, University Paris Sud XI, Orsay Cedex, (France). His main research interests include image analysis and robotic perception systems.

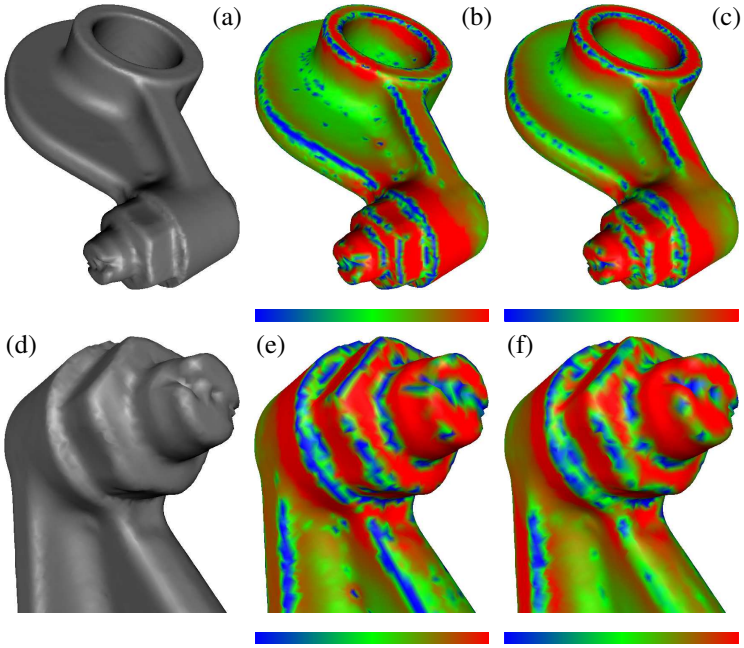


Fig. 1. Rocker arm: example of SOD (b), ESOD (c) and respective zooms (e,f).

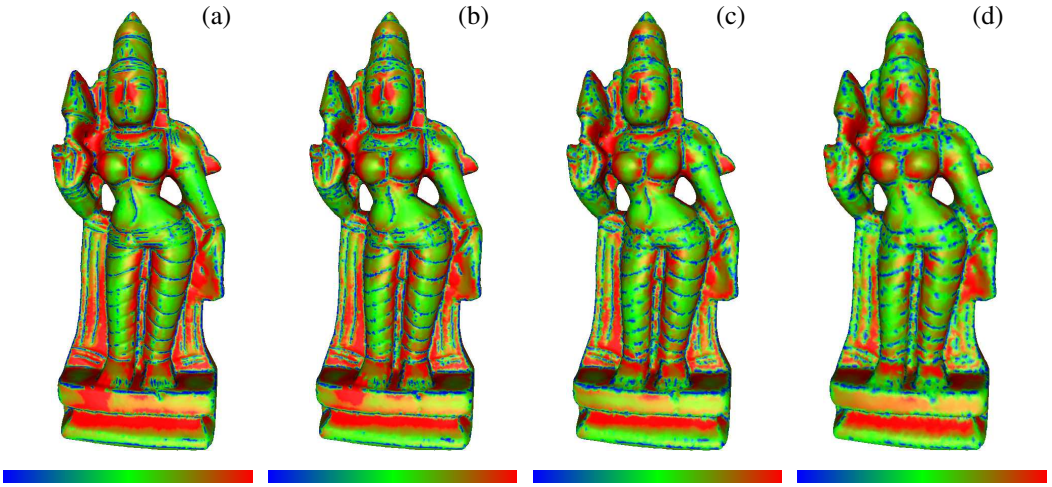


Fig. 2. The proposed algorithm is almost invariant with respect to scale. This example shows the segmentation computed on the full resolution mesh (a) and its reduced versions (b:75% of the original vertices, c:50% and d:25%).

A Multi-Airport Dynamic Network Flow Model with Capacity Uncertainty

Antonio Manca[1] and Paola Zuddas[1]
[1]*University of Cagliari, Cagliari, Italy*
{amanca, zuddas}@unica.it

Abstract—**The objective of Air Traffic Management is to enable aircraft operators to meet their planned times of departure and arrival and adhere to their preferred flight profiles with minimum constraints, without compromising agreed levels of safety. A tactical optimization is a difficult process which requires a practical implementation of the model and the solution strictly depends on the airport capacity uncertainty. It is clear that, the local study of restrictive procedures without considering all the actual system could generate domino effects in other airports. To face this problem it is necessary to analyze the Ground Holding Problem (GHP) under a global aspect and involving all the airports as elements of a system. A global view permits to perform restrictive procedures but in a manner that they are well distributed around all the airports. We modeled the assignment of each airplane by a direct multi-period direct graph, representing a temporal network. Uncertainty in future runway capacity has been managed recurring to scenario analysis. Because this approach produces, lots of scenarios with millions of variables, both grid computing and decomposition methods result to be a valid approach to obtain robust solutions rapidly**.

I. INTRODUCTION

Air traffic delays occur for a variety of reasons ranging from bad weather conditions, requiring runway closure, de-icing, cross-wind limitation (nobody's responsibility), to technical/operational problems such as departure aircraft operations (aircraft operator's responsibility), customer service issues, air traffic control system decisions, equipment failures, airport congestion and lack of Air Traffic Flow Management (ATFM) (i.e.: the inability of certain Air Traffic Control Units to handle all the flights wanting to cross the airspace at a certain time). ATFM system must provide Air Navigation Services to a certain volume of air traffic, satisfying the targeted high level of safety and without imposing significant operational, economic or environmental penalties under normal circumstances. When some unpredictable event occurs, reducing capacity, controllers put in place penalties in order to adapt the demands requests to the actual runway capacity. Examples of operational penalties are: ground delay, assignment of a cruise level different from the optimum one during a long period of flight, extra mileage and holding patterns.

II. LITERATURE REVIEW

Literature reports several decision support for ATFM and it is about two decades old [13]. Research has concentrated on tactical optimisation model based on the US situation.

[11] provides similar European mapping work. In [15] the general flow management is posed, it provides a comprehensive examination of the flow management problem. [17] provides a deterministic linear program formulation for the single airport GHP. Navazio and Romanin-Jacur [14] formulate the deterministic multi-airport GHP with banking constraints as an integer program. Several stochastic models In [10] is discussed how the sensitivity analysis shows a number of limitations and may provide misleading conclusions with respect to the nature of the solutions. Richetta [16] and Richetta and Odoni ([17], [18]) make decisions on a collection of flights. They first solve the probabilistic GHP deterministically by selecting the most likely capacity profile and solving the deterministic linear program. Next, Richetta and Odoni provide a stochastic linear program formulation of the single airport multi-period dynamic stochastic GHP. The resulting stochastic linear program is easily solved by transforming the problem to a minimum cost network flow model. [18], [4], [3] are based essentially on assignment models, they assign a departure time to each flight. Both [6] and [2] give a ground delay model based on dynamic programming that incorporate uncertainty in forecasting capacity. Unfortunately, these models grow exponentially complex with the number of time periods, making them virtually impossible to solve for real-sized problems. [8], [7] model the GHP as a multi-commodity dynamic network flow problem and extend the model including capacity uncertainty. Scenario bundles are grouped in a scenario tree ([21], [19]) and a so called, compath decomposition scheme is proposed where mutual capacity constraints are dualized. [12] takes again the deterministic model presented in [8] and proposes a decomposition scheme, where the non-anticipativity constraints are dualized. The decomposition algorithm in [12] utilizes the bundle method approach presented in [5]. A Scenario analysis approach is also adopted in [1], in which a fix and relax approach has been proposed.

III. NETWORK GENERATION

Given a set Q of different airports, we extend the classical GHP to a set of airports on a given time-horizon T, typically one day. A slot is a time interval assigned to an airplane by controllers in order to permits to airplanes to execute taking off or landing operations under security conditions. Assuming the airplanes have all the same priority, a simple high cost solution can be obtained by assigning the slots with a FIFO (First In First Out) policy. In this case, available slots are assigned to first airplane arrived in the aerodrome airspace and by assigning, in some manner, a ground holding to all airplanes waiting the take off at departure airport. This policy has a possible drawback, that is all costs associated to delays could be not well distributed among all airplanes and this implies a heavy increment of air traffic system costs. Both ground holding and airborne holding operations can be well distributed in different manner by assigning both slots and delays among the airplanes minimizing the total delay cost of the system and exploiting in optimal manner available airport runway capacity. Given a set of airports, we introduce a tactical approach permits to allocate the airplane departures, by assigning, for each flight, a possible both ground holding and airborne holding , minimizing the total delay cost of the system. We assume to subdivide the time horizon T into sequential time steps (periods) of fixed length. We define as Δ_p the size of each period, typically, 15 minutes. This is the interval of time the flight is considered in time. We model the assignment of each airplane delay by a multi-period direct graph, representing a dynamic network, where a number K of airplanes are in competition among them in order to obtain a future slot either to land or to take off. The network is generated starting by associating a double column of indexed nodes to each airport, each column represents the state of an airplane in all periods, see figure 1. To each node i is

associated a time, such that, $t(.): i \Rightarrow I^+$, (I^+: the set of integer positives) returns the associated time. Nodes at same levels in the columns are referred to the same time. Arcs represent change of state and/or change of period. In figure 1, vertical arcs linking two consecutive nodes represent a change of period while the airplane remains in the same state. In this case the airplane maintains the same state in the passage from a period to the successive one, until a slot assignment is obtained. The first column, referred as α-state, represents a waiting state for a slot assignment in all periods. While an airplane is in a α-state, it is waiting for an allocation allowing to perform an operation (i.e. waiting to receive the prompt to allocate in a ready-to-operate state). We assign a high cost to each waiting arc in α-state. If it is referred to first columns, vertical arcs are used to model the waiting state in order to obtain either a permission to landing (holding pattern) or take off in the near proximity of the runway area, such as the taxi-way.

The second column, referred as β-state, represents the ready-to-operate state. In this state the airplane can await a take off permission, can be under maintenance, or is subjected to a penalty of ground holding. We suppose all such operations are performed in parking area. We assign a low cost to each waiting arc in β-state. Given an airport *q,* we define as $\phi_\alpha(q)$, $\phi_\beta(q)$, $q \in Q$, the set of nodes belonging to first (left side) and second (right side) column, respectively

Diagonal arcs represent an airplane state change, from a waiting to an operational state, in this case a slot is assigned to airplane and it can occupy the runway. In this case a state change means a slot is assigned to a generic airplane and the runway resource is available. We assign a very low cost to each diagonal arc. In figure 1, we use two types of oblique arcs: arcs starting from first column toward second column represent the state of slot assignment, arcs starting from a node in second column toward a node in first column model a relocation assignment. For a given airplane, relocation assignment models the event of ready to receive slot availability after the last landing. In fact, after a landing procedure, airplanes have to be subject to a number of compulsory checks and operations such as load/unload operations in parking area, change crew, refuelling etc., in order to prepare the next flight. When airplanes are ready to receive a slot all these operations have to be completed.

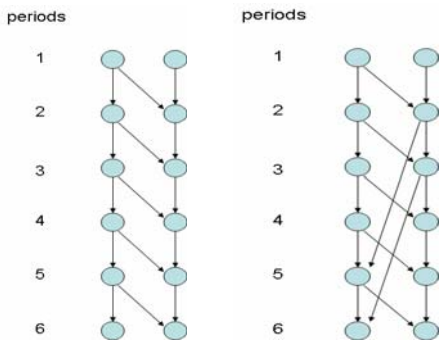


Figure 1: oblique arcs of slot and relocation assignment.

Hence, if it is referred to vertical arcs in second column (figure 1), waiting assignment is used to model airplane additional extra-time in parking area. At this point, each airport belonging to an ATFM system is modeled as a direct graph represented in figure 2.

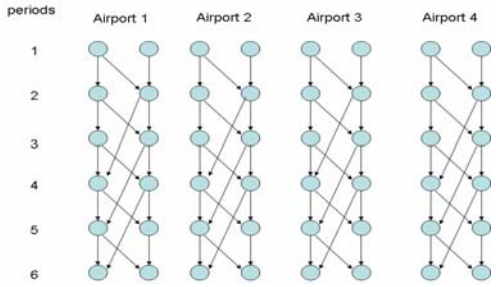


Figure 2: graphs of state for a set of airports.

In figure 3, connections among all airports are represented by oblique arcs; such arcs identify a transaction of state from a given state to a departure airport to another state in a

destination airport. Such arcs start from second column node of graph modeling a departure airport and end to first column node of the graph modeling the landing airport.

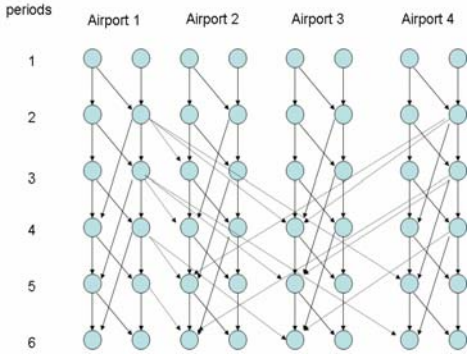


Figure 3: connection arcs among airports.

The difference of level among the starting node and the ending node represents the time to cover the distance among two airports. For each airplane, we have to decide a path from initial departure airport to a final destination airport, each arc in the path can include one or more take off, landing, relocation activity,

$$A_{a,q}=\{(i,j)\in A: i\in\phi_\alpha(q), j\in\phi_\alpha(q), t(i)+\Delta_p=t(j)\} \quad 1.1$$

holding pattern, and ground holding operations. For example in figure 4, it is reported the path *(b,c,d,l,g,m,p)* characterized by bleu arcs where a flight between airport 1 e airport 3 has been performed. Arc (*b,c*) represents the change of state modelling the take off operation, arc (*c,d*) is the change of state representing the flight duration, arc (*d, l*) is the change of state representing a landing operation. Hence, after a relocation phase in airport 2, arc (*l, g*), a slot being assigned at period 7 and there is another flight connection among airport 2 and airport 3, arc (*m,p*). Path *(q,r,f,g,h,n)* characterized by red arcs shows flight connection among airport 4 and airport 2. Arc (*q,r*) represents the change of state modelling the take off operation, arc (*r,f*) is the change of state representing the flight duration, arcs (*f, g*) and (*g, h*) represent a consecutive state of waiting (it takes 2 periods) in order to perform a landing operation. This is the case of a pattern holding because the airplane waits a slot assignment flying on the airport airspace. We can observe that at period 6, both airplanes are in competition among them for a slot to landing and in this example the slot is assigned to flight coming from airport 1. Finally, at period 7 the airplane has been permitted landing and this resource assignment is modeled by arc (*h, n*).

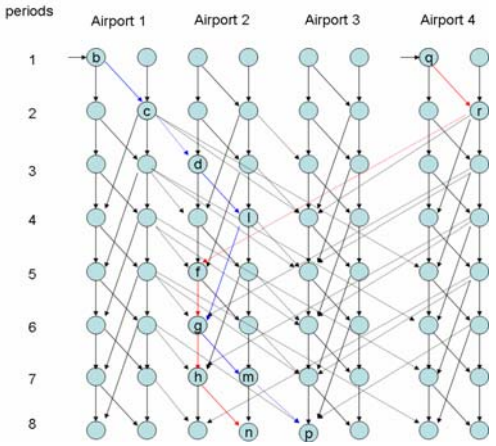


Figure 4: example of different paths.

For each airport *q* we define the following sets:
$A_{a,q}$ is the set of *w*aiting arcs for the admittance permission to runway $\phi_\alpha(q)$. An airplane covering this arc is waiting for landing permission or it is in queue, waiting for a take off operation. A generic element *of* $A_{a,q}$ is indicated as $(i,j)_A$ and both *i* and *j* belongs to the same column of nodes. In case the airplane covers consecutive arcs $A_{a,q}$, this generated delay means either a holding pattern or a ground holding.

$$A_{ar,q}=\{(i,j)\in A: i\in\phi_\beta(q), j\in\phi_\beta(q), t(i)+\Delta_p=t(j)\} \quad 1.2$$

$A_{ar,q}$ is the set of waiting arcs for the admittance in relocating permission. The generic element of $A_{ar,q}$ is indicated as $(i,j)_{ar,q}$ and both *i* and *j* belongs to the same column $\phi_\beta(q)$. A generic arc $(i,j)_{ar,q}$ could represents the delay assigned to a commodity in parking area as subjected to ground holding restriction,

or a programmed halt before the next scheduled flight.

$$A_{r,q} = \{(i,j) \in A : i \in \phi_\beta(q), j \in \phi_\alpha(q), t(i) + l\,\Delta_p = t(j)\} \quad 1.3$$

where l is the number of time-periods requested to perform all compulsory control.

$A_{r,q}$ is the set of arcs corresponding to the relocating activity. These arcs model the mandatory routine procedures of control in parking area. $A_{v,q}$ is the set of arcs representing

$$A_{v,q} = \{(i,j) \in A : i \in \phi_\beta(q), j \in \phi_\alpha(q'), q' \in Q, t(i) < t(j)\} \quad 1.4$$

flights starting from airport q toward any other airport q'.

$A_{va,,q}$ is the set of arcs representing flights arriving in airport q coming from any other airport q'.

$$A_{p,q} = \{(i,j) \in A : i \in \phi_\alpha(q), j \in \phi_\beta(q), t(j) = t(i) + \Delta_p\} \quad 1.5$$

$A_{p,q}$ is the set of arcs corresponding to runway occupation required for either a landing or take off operations. These operations must be completed inside in a time period Δ_p

$$A_{va,q} = (i,j) \in A : i \in \phi_\beta(q'), j \in \phi_\alpha(q), q' \in Q, t(i) < t(j)\} \quad 1.6$$

The final network $G=(N, A)$ includes a set N of nodes, $|N| = n$, and a set A of arcs, $|A| = m$, each element is referred to different states and operations. We adopt the network, $G=(N, A)$ as the topological support to the mathematical model. Starting from the above described sets it is possible to built operative constraints to permit airplane to make rights movements in the network.

IV. Uncertainty Model

Different commodities circulating on the network correspond to the number K of airplanes considered in the system. For each arc $(i,j) \in A$ and a generic commodity $k \in K$. In a scenarios analysis context a tactical strategy which explores all the possible scenarios returning a robust solution is important to aid controllers to take tactical decisions. Scenario analysis approach attempts to face the uncertainty factor by taking into account a set Ω, of different supposed scenarios. Each scenario $g \in \Omega$, corresponds to a different possible time evolution of a single or set of parameters. Each scenario g defers from the others, for the mutual capacity parameters u^g, assigned to runway arcs, and the parameter ω_g. Moreover a so-called weight, ω_g, is assigned to each scenario and represents the "importance" assigned by the manager to the running configuration. Referring to an arc $(i, j) \in A$, a commodity $k \in K$ and a scenario $g \in \Omega$, we adopt the notation: x^g, u^g, c^g to describe respectively, the decision variable, the single capacity and the cost. starting from a given time forward, it occurs to take into account the two possible evolutions in the subsequent diverse parts. In this way, the set of parallel scenarios, figure 5, is aggregated by producing a tree structure, called scenario-tree. In a scenario tree the first deterministic part finishes at time

$$\tau = \max \{u^g = u^{g'} : \forall (g, g') \in \Omega\}$$

the last instant of time arc capacities are equal for all scenarios (see figure 5). In *model 1* uncertainty is described by means of a set of possible future outcomes, the scenarios. At times such weights can be viewed as the probability of occurrence of the examined scenario. More often they are determined on the basis of background knowledge about the system.

Model 1

$$\min \sum_g w_g c_g x^g$$

$$x^* \in S$$

$$x' \in R \quad \forall g \in G$$

$$x^*, x' \text{ in } \{0,1\}$$

where $x^* \in S$, are the congruity constraints and $x' \in S$, with R the operative constraints.

As reported in figure 5, we call as first stage, the deterministic portion of each scenario, left side in figure 5. The second stage represents the interval of time where different hypothesis are formulated about runways capacity and scenario weights, right side in figure 5. A terminal node for each scenario collects all arcs of the scenario guarantees solution admissibility.

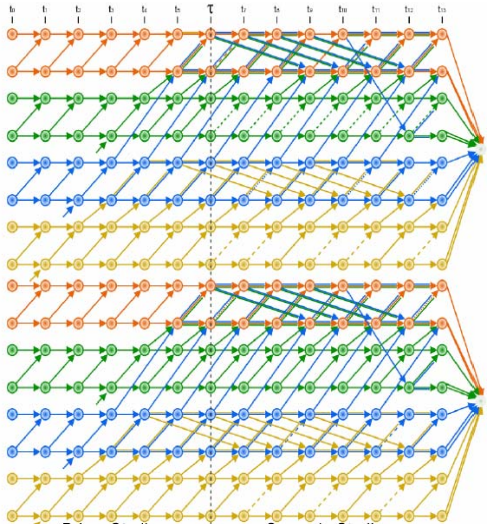


Figure 5: 2 parallel scenarios.

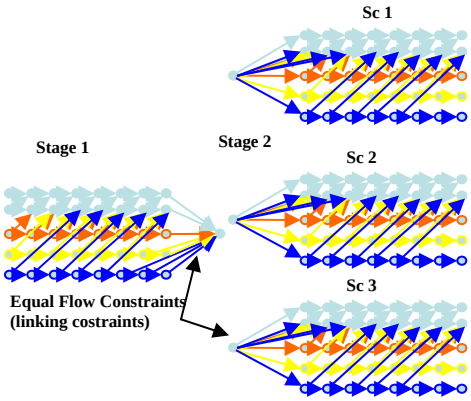


Figure 6: scenario tree decomposition

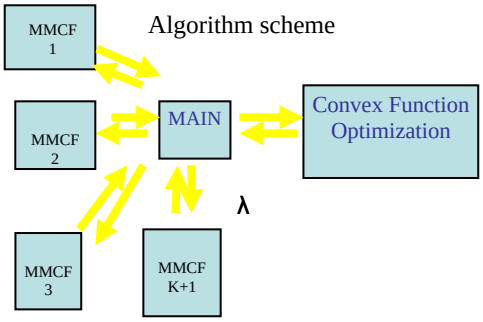


Figure 7: decomposition scheme

V. Decomposition Approach

Scenario tree model can be managed in order to decompose it in sub-models more simple to solve. If linking constraints, corresponding to linking arcs between first and second stage, are relaxed it is possible to obtain a modular structure that can be exploited to build iterative algorithms into Grid computing (figure 6). Given a two stage tree and K scenarios, with this decomposition we have to solve K+1 multi-commodity problems in parallel manner. (figure 7). At each step, obtained solutions are managed by module Main which collects solutions, solves a convex optimisation problem and computes new costs for the MMCF (Minimum-Multi-Commodity-Flow) modules. These steps are repeated until a good convergence has been founded. Combination of open source codes, specialised solvers, modular programming and standardization rules has permitted us to implement a decision support system that can be executed in a grid computing environment. This step has been necessary because the large number of scenarios in a single processor the convergence is very slow also after hours of computing time.

VI. Conclusion

Scenario analysis applied to air traffic delay problems permits us to solve large dimension real-size instances in a short time. We have showed how to decompose a tactical model

based on dynamic networks applied to a set of airports of a total Air Traffic in order to adapt it to GRID computing environment. This approach has permitted to execute independently each multi-commodity problem in different clusters and successively to find an admissible solution for the original problem. It remains some problem to find a robust open source mixed integer problem solver easily adaptable to GRID environment. ATFM model characteristics of the presented model are:

- ground holding and holding pattern assignments;
- multi-period network model;
- scenario analysis under uncertainty in runway capacity.

Relevant information returned after an optization phase is:

- assign delays to balance delays among all airplanes;
- optimize available air traffic resources (capacity);
- plan the path of each airplane period for period;
- minimize the expected total cost of delay;
- domino effects;
- periods airports works near the "hedge".

REFERENCES

[1] Alonso A., L.F. Escudero and M.T. Ortuno. A Stochastic 0-1 Program Based Approach for the Air Traffic Flow Management Problem. European Journal of Operational Research, 2000,12, 47-62.

[2] G. Andreatta, G Romanin-Jacur. Aircraft Flow Management Under Congestion, Tranportation Science, 1987, 21, 249-253.

[3] D. Bertsimas, Odoni A., P.B. Vranas. Dynamic Ground Holding Policies for a Network of Airports, Tranportation Science, 1994, 28, 4, 275-291.

[4] D. Bertsimas, A. Odoni, P.B. Vranas. The Multi-Airport Ground Holding Problem in Air Traffic Control, Operations Research, , 1994, 42, 249-261.

[5] A. Frangioni. Dual-Ascent Methods and Multicommodity Flow Problems,{\it PhD Thesis}, University of Pisa, 1997.

[6] G.D. Glockner. MIDCORT: Minimizing Delay Cost in Real Time, Technical Report, Federal Aviation Administration, AOR-200, Washington, 1991.

[7] G.D. Glockner, Dynamic Network Flow with Uncertain Arc Capacities, PhD thesis, Georgia Institute of Technology, Atlanta GA, 1997.

*[8] G.D. Glockner. Effects of Air Traffic Congestion Delays Under Several Flow Management Policies,Transportation Research Record, 1996, 1517, 29-36.

[9] G.D. Glockner and G.L. Nemhauser. Dynamic Network Flow with Uncertain Arc Capacities: Formulation and Problem Structure,Technical Report, 96-08, Logistics Engineering Center, Georgia Institue of Technology, Atlanta GA.

[10] J. Higle, S. Wallace. Managing Risk in the New Power Business, IEEE Computer Applications in Power, 2002, 12th-19th April.

[11] Leal de Matos P.A., Ormerod R..Application of Operational Research to European air traffic flow management. European Journal of Operational Research, 1999.

[12] Manca A.. Optimization Under Uncertainty: A General View. PhD thesis, University of Cagliari, 1996.

[13] Leal de Matos P.A., Powell P.L.. Decision Support for Flight re-routing in Europe, Decision Support Systems, 2002, 397-412.

[14] L. Navazio, G Romanin-Jacur. The Multiple Connections Multi-Airport Ground Holding Problem: Models and Algorithms, Transportation Science, 1998, 32, 3, 268-276.

[15] A. Odoni, The Flow Management Problem in Air Traffic Problem, in Flow Control of Congested Network, A. Odoni, L. Bianco, G. Szego. Eds. Springer-Verlag, 1987, 269-288.

[16] O. Richetta. Ground Holding Strategies for Air Traffic Control Under Uncertainty, Techincal Report 198, M.I.T., Operations Reasearch Center, 1991.

[17] O. Richetta, A. Odoni. Solving Optimally the Static Ground Holding Policy Problem in Air Traffic Control, Transportation Science, 1993,27:3, 228-238..

[18] O. Richetta,A. Odoni.Dynamic Solution to the Ground Holding in Air Traffic Control, Transportation Science, 28:4, 275-291, 1994.

[19] R.T. Rockafellar and R.J-B Wets.
Nonanticipativity and stochastic optimization problems, Mathematical Programming Study, 1976, 6:170-187.

[20] R.T. Rockafellar and R.J-B Wets. Scenarios and policy aggregation in optimization under uncertainty, Mathematics of Operations Research, 1991, 16:119-147.

[21] R.J-B. Wets.
The aggregation principle in scenario analysis and stochastic optimization. In S. Wallace, editor, Algorithms and Model Formulations in Mathematical Programming, NATO ASI Vol.51, 91-113. Springer, NATO ASI, 1989,Vol.51.

Synthesis of EBG Surfaces Using Evolutionary Optimization Algorithms

L. Deias[1], G. Mazzarella[1] and N. Sirena[1, 2]
[1]*DIEE – Department of Electrical and Electronic Engineering, University of Cagliari*
p.zza d'Armi, 09123 Cagliari, Italy
{luisa.deias, mazzarella}@diee.unica.it
[2]*Consorzio COSMOLAB, via Marengo 3, 09123 Cagliari, Italy*
nico.sirena@gmail.com

***Abstract* — In this paper evolutionary computation is applied to the synthesis of planar periodic EBG. We constrained our evolutionary design to the unit cell geometry and used a full–wave MoM to evaluate all individuals. Results show that the resonant frequency is able to approach the desired value, improving the bandwidth.**

***Index Terms*— Cybersar, EBG, Automatic Design, Genetic Programming, Grid.**

I. Introduction

FREQUENCY selective surfaces (FSS), i.e. periodic metal patches printed on a dielectric substrate, have a long history, because they have been studied since the early 60's both in the microwave and millimeter-wave region [1][2][3]. In the past decade, the study of FSS has received a new strong impulse by the introduction of the electromagnetic bandgap (EBG) structures [4][5]. This entirely new class of structures, encompassing FSS as one of its subclasses (planar EBG), were named in analogy to the band gaps present in electric crystals and present some very interesting new electromagnetic properties. Artificial magnetic conductors (AMC) are promising applications of electromagnetic bandgap (EBG) metamaterials. By choosing the proper geometry of the periodic surface we can shape the electromagnetic behavior of EBGs structures in order to prevent the propagation of electromagnetic waves in a given frequency band. As such, they have been proposed for use in low-profile antenna systems. AMC-backed substrates can be used to enhance the radiation efficiency of patch antennas [6][7][8].The main drawback of this strategy is the reduced bandwidth of the complete antenna, since the frequency range over which these EBG surfaces behave as an AMC is usually narrowband and is fixed by their geometrical configuration.

The aim of this paper is to apply Evolutionary Computation (EC) to the synthesis of planar periodic EBGs.

II. EBG Design Using Evolutionary Computation

Evolutionary strategies [9], [10] are algorithms imitating the principles of natural evolution for optimization problems, based on the idea that most real-world problems cannot be handled with binary representations. Unlike genetic algorithms, chromosomes need not be represented by bit-strings and the alteration process includes other "genetic" operators appropriate for the given structure and the given problem.

Representation is a key issue in genetic algorithms [11] because the representation scheme can severely limit the window through which the system "observes" its world.

Evolutionary strategies allow to use a more appropriate , i.e."natural", data structure and specialized "genetic" operators, thus introducing problem–specific knowledge into the representation.

While the evolutionary strategy is almost standard for a variety of problems, three

elements must be defined to evolve a design, i.e. "individual":

- representation scheme: definition of the design space;
- variation operators: code that takes one or more design representations as input and outputs a design derived from them;
- fitness function: a function that evaluates a design.

In other words, representation refers to a form of data structure. Variation operations are applied to existing solutions to create new solutions. Usually, variation is based on random perturbation. We used three genetic operators: reproduction, mutation and crossover.

The fitness criteria measure the quality of any given solution. The selection method uses the score obtained for each solution to determine which to save and which to eliminate from the population at each generation. Those solutions that survive are described as parents. The initialization of an evolutionary algorithm can be completely at random, or can incorporate human or other expertise about solutions that may work better than others.

The fitness function employed by the EC optimization is:

$$FF = a * b * Phase\,(f_0) \qquad (1)$$

The fitness function quantifies the optimality of a solution focusing in our case mainly on the center frequency, i.e. the last term of (1), which becomes small when the resonant frequency is close to the desired working frequency. Then two penalty coefficients are added, a and b, respectively developed in order to avoid geometries with high number of discretization elements within the unit cell and favour individuals with larger bandwidth.

A full–wave simulation using Method of Moments is carried out on a single design, i.e. unit cell of the infinite periodic surface and individual of a set population. Starting point is a random initial population of individuals. The external parameters of the structure (i.e. substrate thickness and dielectric constant) are fixed, altogether with the periodicity of the planar EBG. The geometry of the aperture is subject to the EC optimization, while the unit square cell within which each individual/design can evolve is fixed together with its discretization step. The set of admissible solutions is then composed by every geometry/aperture that can be designed in this square and built as series of segments that can evolve in every direction, with no limit on the number of segments, on their width and length and number of subsequent ramifications. We fixed the discretization step within the unit cell in order to pose a limit not only to the computational burden but also to the geometrical spatial bandwidth of the possible solutions. In such a way we gain a stabilizing effect on the problem, much as shown in [12] for a different problem, namely wire antennas.

The phase of the reflection coefficient, computed at one or more frequencies depending on the goal is then incorporated into the EC strategy for the evaluation procedure, the fitness function determining the environment within which the solutions "live" and the best individual of a generation. The evolutionary operators (reproduction, crossover and mutation) are then applied to the best individual leading to the subsequent evolution of the population and the next generation. In this way the geometrical properties are optimized to evolve the best solution (zero phase at the desired resonant frequency).

III. GRID IMPLEMENTATION

A grid implementation has been developed with a Master-Slave architecture. The master, running on the User Interface, builds the individuals of each generation, delegating to the slaves the burden of their evaluation. Each slave is then submitted in the grid as a job and,

once started, a TCP/IP connection is opened I order to exchange data with the master. To each slave is associated a queue of individuals waiting for analysis and when a slave finishes an evaluation a new individual is inserted in its queue. By doing so the evaluation process (by far the most expensive) can be divided into N machines, thus obtaining an increase in performance proportional to N.

The master, using Java API, is able to automatically submit new jobs/slaves and/or terminate those in execution, so that the user acting on specific controls of the GUI can dynamically vary the number of active slaves. Currently we are working on a mechanism, based on a continuous monitoring of available resources (Worker-Nodes), that could automatically adjust the number of slaves submitted in order to optimise the resource utilization without disturbing other simulations.

IV. RESULTS

We considered as reference structure the well–known geometry [4] shown in Fig.1, consisting of a Jerusalem cross aperture. The reference structure has periodicity $d_x = d_y =$ 3.048 mm, h = 0.635mm and $\varepsilon_r = 10.2$, displaying a resonant frequency at 14.25GHz in the uncovered configuration. The reflection coefficient is shown in Fig.2.

We used for the full–wave analysis of the periodic structure our previously developed software, based on an efficient aperture approach to the analysis of single and multi–layer periodic AMCs [13]. Furthermore this approach is irrespective of the particular shape of the metallization, which in the EC optimization stage cannot be known in advance.

The evolutionary strategy implementation chosen was ECJ, a general purpose Java-based Evolutionary Computation research system developed at George Mason Universitys ECLab [14]).

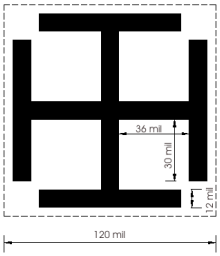


Fig.1. Jerusalem Cross aperture.

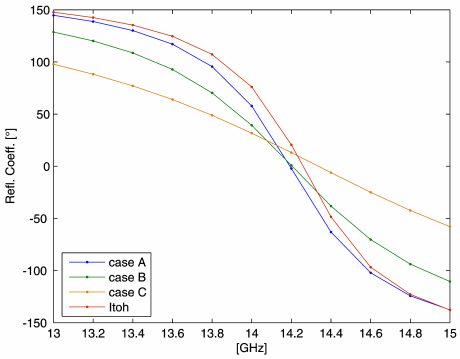


Fig.2. Reflection Coefficient.

The EC optimization was launched to find the best structure with only restrictions represented by the periodicity, and substrate dielectric constant and thickness, which were our fixed parameters. In Fig.3 we can see three different runs of our optimization and in Tab.I it is shown how the goal is achieved, in comparison also to the reference structure.

In Case A we used for the unit cell a discretization step of 0.1016mm, making the playground for the EC optimization a 30x30 grid. The penalty coefficients, both for the number of unknowns (i.e. borders in the discretization, which are shown in red) and for the bandwidth were not too strict, making the evolution process quickly evolve towards quite bulky structures. The resonant frequency is achieved after not so many generations (for case A at gen. 13 the phase of the reflection coefficient is 2.2◦).

In Case B we acted more stringently on the penalty coefficients. The structure remains spatially limited, as the number of unknowns,

and the bandwidth is largely improved.

	14GHZ	14.2GHZ	14.4GHz
Itoh (lit)	76.05°	20.52°	-48.56°
Case A	57.8°	2.2°	-63.16°
Case B	39.28°	0.83°	-38.16°
Case C	13.06°	-6.13°	-24.93°

Table I

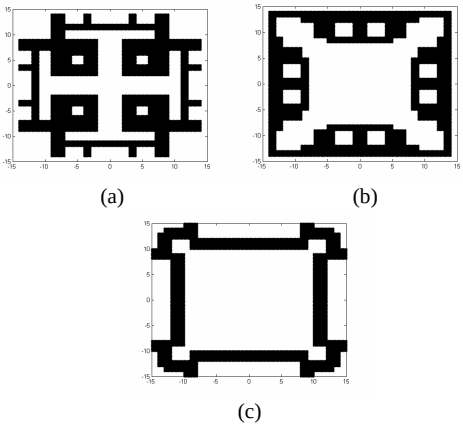


Fig.3. Optimization Runs.

The number of generations in order to achieve a good result is still low.

As expected, the geometries resemble the reference one and other well–known configurations, since the only stringent requirement in this case is the resonant frequency. The fitness in this case, than can be seen as a test case, is actually relatively simple, aimed almost uniquely to direct the evolution process towards a structure that resonates at the desired frequency. We then can insert an additional penalty coefficient in order to maximize the bandwidth, a secondary objective overlapped to the main goal.

Finally, in case C, we allowed the geometry to touch the borders of the unit cell, thus considering a continuous aperture (while the geometry of the metallic patch will therefore be finite), and we found out a further improvement in the bandwidth.

We also tried different discretization steps within the unit cell and, as expected, we found out that when becoming too small it can lead to an increased probability for the fitness to generate local minima/maxima, which trap the evolutionary process.

V. Conclusions

Surface EBGs display many drawbacks when used in conjunction with other planar structures (i.e. antennas), while the possible choice of geometries is limited or when using standard genetic algorithm hardly realizable.

The main advantage of the genetic approach is the generalization of the solution space, evolution being able to breed geometries that we could never think of. Evolutionary computation on the other hand let us scope through geometries as though they were human–described.

The EC strategy will allow new geometries, less intuitive and "obvious", to evolve and emerge when considering more complicated structures (i.e. multilayer, with different geometries and substrate parameters) and more strict goals.

Results show that the resonant frequency is able to approach the desired value in a reasonable number of generations, evolution being able to widen the bandwidth. These optimized artificial materials can lead the way to the development and design of innovative electromagnetic structures.

Acknowledgments

This work makes use of results produced by the Cybersar Project managed by the Consorzio COSMOLAB, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Programma Operativo Nazionale 2000-2006 "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" per le Regioni Italiane dell′Obiettivo 1 (Campania, Calabria, Puglia, Basilicata, Sicilia, Sardegna) - Asse II, Misura II.2 "Società dellInformazione", Azione a " Sistemi di

calcolo e simulazione ad alte prestazioni". More information is available at http://www.cybersar.it

REFERENCES

[1] R. Mittra, C. H. Chan, and Y. Cwik, "Techniques for analyzing frequency selective surfaces – a review," in Proc. IEEE, Dec. 1988.

[2] A. Munk, Frequency Selective Surfaces: Theory and Design. New York: Wiley Interscience, 2000.

[3] T. K. Wu, Frequency Selective Sur Array. New York:Wiley, 1995.

[4] F. R. Yang, Y. Q. K. P. Ma, and T. tem waveguide using uniplanar co bandgap (uc-pbg) structure," IEEE Trans.Microwave Theory Tech., vol. MTT–47, no. 11, pp. 2092–2098, Nov.1999.

[5] D. Sievenpiper, R. B. L. Zhang, N. Alexopolous, and E. Yablonovitch, "High-impedance electromagnetic surfaces with a forbidden frequency band," IEEE Trans. Microwave Theory Tech., vol. MTT–47, no. 11, pp. 2059–2074, Nov. 1999.

[6] A. Feresidis, G. Goussetis, S. Wang, and J. Vardaxoglou, "Artificial magnetic conductor surfaces and their application to low-profile highgain planar antennas," IEEE Trans. Antennas Propagat., vol. 53, no. 1, pp. 209–215, Jan. 2005.

[7] R. Gonzalo, P. de Maagt, and M. Sorolla, "Enhanced patch-antenna performance by suppressing surface waves using photonic-bandgap substrates," IEEE Trans. Microwave Theory Tech., vol. MTT–47, no. 11, pp. 2131–2138, Nov. 1999.

[8] M. Hosseini, A. Pirhadi, and M. Hakkak, "Design of a novel amc with little sensitivity to the angle of incidence and very compact size," in IEEE Antennas and Propagation Society International Symposium, July 9–14, 2006.

[9] D. B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, 3rd ed. New York: IEEE Press, 2006.

[10] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs. Springer–Verlag, 1992.

[11] J. R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection. Mit Press, 1992.

[12] R. E. Collin, Antennas and Radiowave Propagation. McGraw–Hill, 1985.

[13] F. Asole, L. Deias, and G. Mazzarella, "A flexible full–wave analysis of multilayered amc using an aperture oriented approach," J. Electrom. Waves Appl., vol. 21, no. 14, pp. 2059–2072, 2007.

[14] Ecj website. [Online]. Available: http://cs.gmu.edu/ eclab/projects/ecj/

PLACE PHOTO HERE

Luisa Deias. Was born in Cagliari, in 1972. She received the Laurea Degree in Electronic Engineering in 2002 and the Ph.D in Electronic Engineering and Computer Science in 2005 from the University of Cagliari. She is currently working at the Department of Electrical and Electronic Engineering of the University of Cagliari. Her current research include electromagnetic bandgap structures, artificial magnetic conductors, antenna design and electromagnetic modeling techniques.

Giuseppe Mazzarella (M'85–SM'90) was born in Napoli, in 1959. He received the Laurea Degree in Electronic Engineering summa cum laude in 1984 and the Ph.D. in Electronic Engineering and Computer Science in 1989 from the Università Federico II in Napoli. From 1990 to 1992 he was assistant professor in the department of Electronic Engineering of the Università Federico II in Napoli. In 1992 he joined the Department of Electrical and Electronic Engineering of the Università di Cagliari where he is now full professor of Electromagnetic Fields and Antennas theory and design. His current research interests are in the field of antenna pattern synthesis, analysis and design of patch and slot antennas and inverse problems in electromagnetics.

Nicola Sirena was born in Cagliari, in 1972. He received the Laurea Degree in Electronic Engineering in 2007 from the Università di Cagliari. Since June 2007 he works in collaboration with the Department of Electrical and Electronic Engineering of the Università di Cagliari. His current research interests are in the field of automated antenna design.

Grid-based Atherosclerosis Simulations

Francesco Pappalardo[1], Marzio Pennisi[1], Emilio Mastriani[2] and Santo Motta[2]
[1] University of Catania, Catania, Italy, {francesco,mpennisi,motta}@dmi.unict.it
[2] Consorzio COMETA, Catania, Italy, mastriani@dmi.unict.it

***Abstract*—Atherosclerosis, a pathology affecting arterial blood vessels, is one of most common disease of the developed countries. We present studies on the increased atherosclerosis risk using an agent based model of atherogenesis that has been previously validated using clinical data. It is well known that the major risk in atherosclerosis is the persistent high level of low density lipoprotein (LDL) concentration. However, it is not known if short period of high LDL concentration can cause irreversible damage and if reduction of the LDL concentration (either by life style or drug) can drastically or partially reduce the already acquired risk. We used the Sicilian Grid infrastructure to simulate three different clinical situations in a large set of virtual patients (200 per clinical scenario). In the first one the patients lifestyle maintains the concentration of LDL in a no risk range. This is the control case simulation. In the second and in the third simulations, the lifestyle of the virtual patients raises the LDL concentration to a risk level. Differences in the foam cells formation can be interpreted as permanent or non-permanent risk effects. Finally we consider virtual patients whose life style raises many times the level of LDL concentration just above the normal but this is quickly reduced using appropriate treatment. Those preliminary results obviously need to be clinically investigated. The Grid power allowed us to retrieve results of the simulation in a short time.**

***Index Terms*—Agent based model, atherosclerosis, Grid, HPC.**

I. Introduction

ATHEROSCLEROSIS, a disease affecting arterial blood vessels, is one of most common disease of the developed countries. It is, in large part, due to the deposition of low density lipoproteins (LDLs), i.e., plasma proteins carrying cholesterol and triglycerides, that determine the formation of multiple plaques within the arteries [1], [2]. The origin of atherosclerosis is still not fully understood. However there are risk factors which increase the probability of developing atherosclerosis in humans. Some of these risk factors are beyond a person's control (smoking, obesity), others seem to have genetic origin (familial hypercholesterolemia, diabetes, hypertension) [3]. Common denominator in all the form of atherosclerosis is the elevated level of LDL, which is subject to oxidation becoming ox-LDL, that promotes an inflammatory response and immune activation in the artery walls [4]. The formation of atherosclerotic plaques in the artery reduces both the internal diameter of vessels and the blood flux leading to a number of serious pathologies [5]. Early studies demonstrated that ox-LDL can induce activation of monocytes/macrophages, endothelial cells and T cells. Ox-LDLs engulfed by macrophages form the so called foam cells [6]. These cells represent the nucleus of the plaques formation. Ox-LDL promotes also immune activation of B cells inducing the production of specific anti ox-LDL antibody (OLAB).

Atherosclerosis and their anatomical consequences cause severe problems. Stenosis (narrowing) and aneurysm of the artery are chronic, slowly progressing and cumulative effects indicating the progression of atherosclerotic disease. In both case the result is an insufficient blood supply to the organ fed by the artery. Most commonly, soft plaque suddenly ruptures, causes the formation of a thrombus that will rapidly slow or stop blood flow, leading to death of the tissues fed by the artery. This catastrophic event is called infarction and is not predictable. The most common event is thrombosis of the coronary artery causing infarction (a heart at-

tack): However, since atherosclerosis is a body wide process, similar events also occur in the arteries of the brain (stroke attack), intestines, kidneys, etc. Those atherosclerosis associated events often cause of dead or serious invalidate diseases and require preventive treatments. Vaccine research for atherosclerosis is a hot pharmaceutical topic.

Recently we proposed a model based on the Agent Based Model (ABM) paradigm [7] which reproduces clinical and laboratory parameters associated to atherogenesis. The model and its computer implementation (SimAthero simulator) considers all the relevant variables that play an important role in atherogenesis and its induced immune response, i.e., LDL, ox-LDL, OLAB, chitotriosidase and the foam cells generated in the artery wall.

In this paper we will analyze three different situations over a time scale of two years. The standard normal patients where no foam cells are formed; patients having high level of LDL but who delay to apply appropriate treatments and finally patients who may have many events of high level of LDL but takes immediately appropriate treatments.

The plan of the paper is the following. In §II we briefly describe the model of the Immune control of atherogenesis; in §III we describe our simulated patients and show the simulator results. We briefly draw conclusions and future extension of this work in §IV.

II. **Description of the model**

*a) **The biological scenario.**:* Exogenous and endogenous factors induce in humans a very small, first oxidative process of blood circulating native LDLs (minimally modified LDLs or mm-LDLs). In endothelium mm-LDLs are extensively oxidized from intracellular oxidative products and then recognized by the macrophage scavenger receptor. High level and persistent in time LDLs lead to macrophages engulfment and their transformation in foam cells. Contrary, low level of LDLs and their oxidized fraction, lead to the internalization of the oxidized low density lipoproteins and subsequent presentation by major histocompatibility complex class II at the macrophages surface. Recognition of ox-LDL by macrophages and naive B cells, leads, by T helper lymphocytes cooperation, to the activation of humoral response and production of OLAB. When the OLAB/ox-LDL immune complexes are generated in the vascular wall, the macrophages catch them by the Fc receptor or via phagocytosis and destroy ox-LDL in the lysosome system. During this process, the activated macrophage releases chitotriosidase enzyme, that is then used as a marker of macrophage activation.

*b) **The Model.**:* To describe the above scenario one needs to include all the crucial entities (cells, molecules, adjuvants, cytokines, interactions) that biologists and medical doctors recognize as relevant in the game. The model described in [7] contains entities and interactions which both biologist and MD considered relevant to describe the process.

Atherosclerosis is a very complex phenomenon which involves many components some of them not fully understood. In the present version of the simulator we considered only in the immune system processes that control the atherogenesis. These processes may occur in immune system organs like lymph nodes or locally in the artery endothelium. To describe the Immune processes we considered both cellular and molecular entities.

Cellular entities can take up a state from a certain set of suitable states and their dynamics is realized by means of state-changes. A state change takes place when a cell interacts with another cell or with a molecule or both of them. We considered the relevant lymphocytes that play a role in the atherogenesis-immune system response, B lymphocytes and helper T lymphocytes. Monocytes are represented as well and we take care of macrophages. Specific entities involved in atherogenesis are present in the model: low density lipoproteins, oxidized low density lipoproteins, foam cells, auto antibodies anti oxidized low density lipoproteins and chitotriosidase enzyme. Cytotoxic T lymphocytes

are not taken into consideration because they are not involved in the immune response (only humoral response is present during atherogenesis).

Molecular entities The model distinguishes between simple small molecules like interleukins or signaling molecules in general and more complex molecules like immunoglobulins and antigens, for which we need to represent the specificity. We only represent interleukin 2 that is necessary for the development of T cell immunologic memory, one of the unique characteristics of the immune system, which depends upon the expansion of the number and function of antigen-selected T cell clones. For what is related to the immunoglobulins, we represent only type IgG. This just because at the actual state we don't need to represent other classes of Ig and because IgG is the most versatile immunoglobulin since it is capable of carrying out all of the functions of immunoglobulins molecules. Moreover IgG is the major immunoglobulin in serum (75% of serum Ig is IgG) and IgG is the major Ig in extra vascular spaces.

The actual model does not consider multicompartments processes and mimics all processes in a virtual region in which all interactions take place. Our physical space is therefore represented by a 2D domain bounded by two opposite rigid walls and left and right periodic boundaries. This biological knowledge is represented using an ABM technique. This allows to describe, in a defined space, the immune system entities with their different biological states and the interactions between different entities. The system evolution in space and in time is generated from the interactions and diffusion of the different entities. Compared to the complexity of the real biological system our model is still very naive and it can be extended in many aspects. However, the model is sufficiently complete to describe the major aspects of the atherogenesis-immune system response phenomenon.

The computer implementation of the model (SimAthero hereafter) has two main classes of parameters: the first one refers to values known from standard immunology literature [8], [9] [10], [11]; the second one collects all the parameters with unknown values which we arbitrarily set to plausible values after performing a series of tests (*tuning phase*).

The simulator takes care of the main interactions that happens during an immune response against atherogenesis.

Physical proximity is modeled through the concept of lattice-site. All interactions among cells and molecules take place within a lattice-site in a single time step, so that there is no correlation between entities residing on different sites at a fixed time. The simulation space is represented as a $L \times L$ hexagonal (or triangular) lattice (six neighbors), with periodic boundary conditions to the left and right side, while the top and bottom are represented by rigid walls. All entities are allowed to move with uniform probability between neighboring lattices in the Grid with equal diffusion coefficient. In the present release of the simulator chemotaxis is not implemented.

LDLs values can be fixed in order to simulate different patients both in normolipidic condition and in hypercholesterolemic condition. The same applies to ox-LDLs. However human habits change with time and personal life style. A normolipidic patient can change its attitude becoming an hypercholesterolemic one and vice versa. For this reason we allow the simulator to accept varying life style conditions and preventive actions to decrease risk factors.

III. **Results**

The model described include the possibility of mimicking biological diversity between patients. The general behavior of a class of virtual patients arise from the results of a suitable set of patients, i.e., the mean values of many runs of the simulator of different patients under the same conditions. The class of virtual patients described by the model were tuned against *human* data data collected by [12], [13] where different conditions, normal and hypercholesterolemic diabetic patients were analyzed.

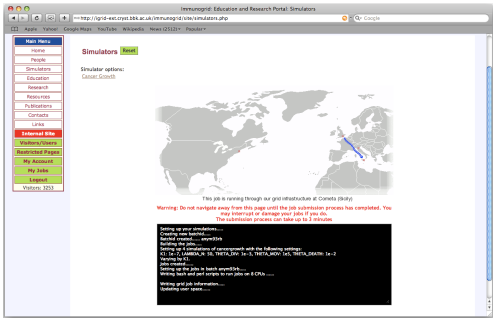


Fig. 1. The submission process of Grid jobs to COMETA infrastructure using ImmunoGrid web portal

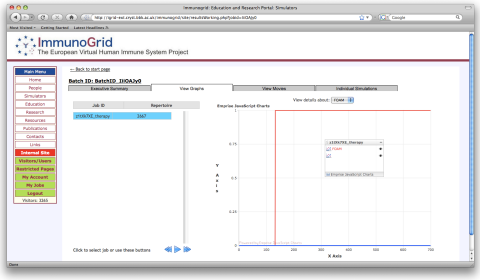


Fig. 2. Simulation results of a virtual patient with level of LDL considered normal. The follow-up period is two years. The figure shows that foam cells formation is absent in this patient.

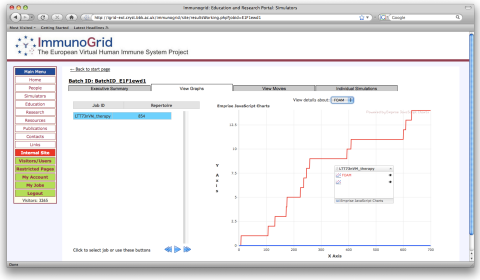


Fig. 3. Simulation results of a virtual patient with level of LDL considered at high risk. The follow-up period is two years. The figure shows that foam cells formation is present, leading to an atherogenesis process.

In this section we analyze the behavior of the same patients in three broad class of clinical conditions to show how SimAthero could be used in order to analyze and predict the effects of various LDL levels in blood. The normal patient simulation is used as control experiment for the other simulations. The differences among these four clinical conditions depend on the LDL level and the time interval which occurs between the time in which concentration of LDL rise above normal level and the time in which the patient take appropriate treatments (lifestyle o drug) to reduce it to normal level.

Jobs were launched using the SimAthero simulator on the COMETA Grid. The submission process was done through the web interface of the ImmunoGrid project (http://www.immunogrid.eu). Figure 1 shows the submission process.

A patient with a LDL level of roughly 950-970 ng/μl of blood is considered normal in clinical practice and he has with very low risk of atheroslerotique plaque. The results of SimAthero for a virtual normal patient (Figure 2) show that he will not support the formation of foam cells and, as a consequence, the beginning of atherogenesis process is absent.

We then simulated a scenario in which a patient, due to several reasons (diet, life style, oxidative agents and so on, so forth) leads its LDL level at 1300 ng/μl, taking it up to 1700 ng/mul. Looking at figure 3 one can observe about 12 foam cells per μl at the end of in silico follow up. This leads to a small atherogenesis process due to the high level of LDL.

Lastly (figure 4), we analyzed a virtual patient that initially takes its LDL level to small peaks, causing no damage. After that, he takes its LDL level to a hypercholesterolemic behavior, generating a small damage, as shown. This shows that small LDL alteration are completely taken under control by the normal behavior of the organism, but high LDL peaks lead to foam cells formation and then to the beginning of the atherogenesis process.

IV. **Conclusions**

Atherosclerosis is a pathology where the immune control plays a relevant role. In this article, we presented studies on the increased atherosclerosis risk using an ABM model of

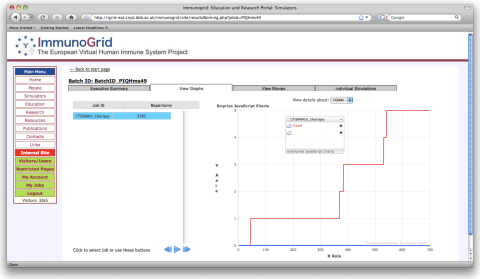


Fig. 4. Simulation results of a virtual patient with level of LDL considered quasi-normal at the beginning and then at high risk. The follow-up period is two years. The figure shows that foam cells formation is negligible in the first time, but becomes important soon after.

atherogenesis and its induced immune system response in humans. Very few mathematical models [14], [15] and (to our best knowledge) no computational models of atherogenesis have been developed to date.

It is well known that the major risk in atherosclerosis is persistent high level of LDL concentration. However it is not known if short period of high LDL concentration can cause irreversible damage and if reduction of the LDL concentration (either by lfe style or drug) can drastically or partially reduce the already acquired risk.

Using an ABM cellular model describing the initial phase of plaque formation (atherogenesis) we are able to simulate the effect of life style which increases the risk of atherosclerosis.

The Cometa Grid infrastructure allowed us to launch hundreds of jobs and obtain results in a reasonable time.

Acknowledgments

This work was supported under the EC contract FP6-2004-IST- 4, No.028069 (ImmunoGrid). This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at: http://www.pi2s2.it and http://www.consorzio-cometa.it.

References

[1] Hanson, G.K. (2002) Inflammation, atherosclerosis, and coronary artery disease, *N. Engl. J. Med.*, **352(16)**, 1685-1695.

[2] Ross,R. (1999) Atherosclerosis–an inflammatory disease, *N. Engl. J. Med.*, **340(2)**, 115-126.

[3] Romero-Corral,A., Somers,V.K., Korinek,J., Sierra-Johnson,J., Thomas,R.J., Allison,T.G., Lopez-Jimenez,F. (2006) Update in prevention of atherosclerotic heart disease: management of major cardiovascular risk factors, *Rev. Invest. Clin.*, **58(3)**, 237-244.

[4] Berliner,J.A., Heinecke,J.W. (1996) The role of oxidized lipoproteins in atherogenesis, *Free Radic. Biol. Med.*, **20(5)**, 707-727.

[5] Vinereanu,D. (2006) Risk factors for atherosclerotic disease: present and future, *Herz*, **Suppl. 3**, 5-24.

[6] Steinberg,D. (1997) Low density lipoprotein oxidation and its pathobiological significance, *J. Biol. Chem.*, **272(34)**, 20963-20966.

[7] Pappalardo,F., Musumeci, S., Motta, S. (2008) Modeling immune system control of atherogenesis, *Bioinformatics*, **24:15**, 1715-1721.

[8] Goldsby,R.A. et al. (2000) In Austen,K.F., Frank,M.M., Atkinson,J.P. and Cantor,H. (eds.) Kuby Immunology. W.H. Freeman and Company, New York.

[9] Klimov,A.N., Nikul'cheva,N.G. (1999) Lipid and Lipoprotein Metabolism and Its Disturbances, St. Petersburg: Piter Kom.

[10] Abbas,A.K., Lichtman,A.H., Pillai,S. (2007) Cellular and molecular immunology, *Saunders*, 6th edition.

[11] Celada,F., Seiden,P.E. (1996) Affinity maturation and hypermutation in a simulation of the humoral immune response, *Eur. J. Immunol.*, **26**, 1350.

[12] Brizzi,P., Tonolo,G., Carusillo,F., Malaguarnera,M., Maioli,M., Musumeci,S. (2003) Plasma Lipid Composition and LDL Oxidation, *Clin. Chem. Lab. Med.*, **41(1)**, 56-60.

[13] Brizzi,P., Tonolo,G., Bertrand,G., Carusillo,F., Severino,C., Maioli,M., Malaguarnera,L., Musumeci,S. (2004) Autoantibodies against oxidized low-density lipoprotein (ox-LDL) and LDL oxidation status, *Clin. Chem. Lab. Med.*, **42(2)**, 164-170.

[14] Cobbold,C.A., Sherratt,J.A., Maxwell,S.R.J. (2002) Lipoprotein Oxidation and its Signicance for Atherosclerosis: a Mathematical Approach, *Bulletin of Mathematical Biology*, **64**, 65-95.

[15] Ibragimov,A.I., McNeal,C.J., Ritter,L.R., Walton,J.R. (2005) A mathematical model of atherogenesis as an inflammatory response, *Math Med Biol.*, **22(4)**, 305-333.

Francesco Pappalardo earned his master and PhD. degree from University of Catania, in 2000 and 2004 respectively. Actually he is holding a postdoctoral fellow and he is temporarily assigned to the professorship of Mathematics at the University of Catania. From 2001 to 2004 he did research in computer operating system security. In 2008 he was visiting scholar at Dana-Farber Cancer Institute in Boston (USA) while in 2006 he visited the Molecular Immunogenetics Labs, IMGT in Montpellier (France). His major research areas are on modeling human and mouse immune system responses involved in several pathologies, including viral infections, tumors, and atherosclerosis. This effort leaded and actually leads to both a better understanding of the immune system, and to the application of these modeling approaches to vaccine research. Up to now, Francesco Pappalardo published more than 40 reviewed research papers. He serves the scientific community as editorial board member, reviewer and program committee member for major journals and conferences in the area of computer science and computational biology.

Santo Motta was born in Rome in 1946. He obtained "Laurea" in Physics at the University of Catania (1970) and its M.Sc. in Applied Mathematics at the University of London, Queen Mary College (1971). Actually he is associate professor of Mathematical Physics at the Faculty of Pharmacy of the University of Catania and he is member of the Dept. of Maths & Computer Science of the University of Catania. Presently he is interested in computational models of the cancer immune system competition induced by an in vivo tested immune prevention vaccine. Its present scientific interests are BioMaths, BioComputing and BioInformatics, Immunomics. Previously he was interested in numerical methods for transport equation, Non linear waves in classic and relativistic fluid?dynamics, General Relativity and Cosmology, Solar Physics. Up to now, Santo Motta published more than 80 reviewed research papers and he is member of editorial board and reviewer of several international journals.

Marzio Pennisi graduated in Computer Science in March, 2006, at University of Catania, Italy. In December of the same year, he won the competition for Ph.D. course of study at the University of Catania, Italy. He is actually studying models, mathematical methods and computer simulation of the immune system. In particulary he is working on models of mammary tumor growth, diffusion of metastasis and therapy. His present scientific interests range from BioMaths BioComputing and BioInformatics Immunomics, evolutive algorithms, constrained and unconstrained optimization.

A Biological Optimization Problem on the Grid

Marzio Pennisi[1], Francesco Pappalardo[1], Emilio Mastriani[2], Ferdinando Chiacchio[3] and Santo Motta[1]

[1]*University of Catania, {mpennisi,francesco,motta}@dmi.unict.it*

[2]*COMETA, COnsorzio Multi Ente per la promozione e l'adozione di Tecnologie di calcolo Avanzato, Catania, mastriani@dmi.unict.it*

[3]*Faculty of Engineering, University of Catania, chiacchio@dmi.unict.it*

***Abstract*—Vaccine and drug discovery is a long process. Mathematical-computational models can speed-up the process. In cancer immunology the Triplex vaccine [1] prevented mammary carcinoma formation in HER-2/Neu mouse models using a *Chronic Schedule*, but it is not known if this schedule is minimal.**

The use of the SimTriplex model and Genetic Algorithms produced a minimal vaccination schedule capable of guarantee *in silico* survival [1] requiring an HPC infrastructure for several days.

In this paper we show a novel optimal protocol search algorithm based on Simulated Annealing (SA) capable to greatly reduce the computational effort. The COMETA grid infrastructure has been used for protocol search and validation.

***Index Terms*—COMETA, Drug, Vaccine, Protocol, Immunology, Tumor.**

I. Introduction

THE last decade has been characterized by a vigorous fast innovation along the whole sectors. Computer technologies and the internet grew up rapidly favouring the born of new computational frameworks. Nowadays clusters of computers connected by heterogeneous networks can deliver powerful computational capabilities to face up to previoulsy unresolved problems of the most disparate scientific fields.

Breakthroughs in the fields of genetics, biochemistry, and molecular biology, in conjunction with this technological growth, pushed out the born of a new hybrid discipline called Bioinformatics, whose aim is to improve and extend our biological and immunological knowledge.

Vaccine design and drug discovery is one of the main areas for which this new science can produce a moltitude of new applications, resulting in a great contribute for the research community.

Actually, this activity pushes the following pharmaceutical research and development, (R&D) pipeline [2] which is a lengthy process and involves a severe sequential chain of steps which cannot be shortened.

Mathematical and computational models can help to speed-up this process.

Three potential areas for computational modeling can in fact be developed: *cell signal behaviour* models which aid to understand how compounds affect intracellular signalling, *cell signal-cell behaviour response* models used to correlate intracellular signals to the cell behaviours and *physiological* models, able to simulate the organ level behaviour and predict the system response. Each class of model can help identify new drug targets.

Simulation results coming from computing environments can be compared, through the use of validation tools, with a large amounts of clinical data, organized into ontologies and stored in databases (ie. gene sequencing maps, radiographical exams, medical analysis, etc.). The results of these experiments either further validate the model or identify novel biology that

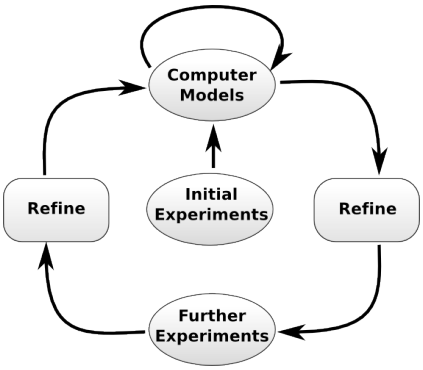


Fig. 1. The cyclical refinement.

is then incorporated into the model. So, it is fair that the development of such kind of simulation environments is an ongoing process which takes into account new knowledge from science and new available technology, achieving the cyclical refinements by means of interdisciplinary efforts (see fig. 1) and opening the scene to what can be considered the *in silico* prediction methods.

Many examples of in silico techniques have been developed to identify suitable vaccine candidates since their use offer a valid filter for the in vitro techniques, more expensive and slower than the former. In [3] there is a wide reference literature of the different fields of application, nowaday in progress. Some other interesting applications of system biology models and optimization techniques can be found in recent literature [3]. Mathematical continuous models have been proposed to describe cancer therapy [4], [5], to optimize immunotherapy [6] and chemotherapy [7].

In this paper we show an example of how a *physiological* model in conjuction with some optimization techniques can be used to speed-up the research of an optmal vaccination schedule for an immunopreventive vaccine.

A vaccination schedule is usually designed empirically using a combination of immunological knowledge, vaccinological experience from previous endeavors, and practical constraints. In subsequent trials, the schedule of vaccinations is then renewed on the basis of the protection elicited in the first batch of subjects and their immunological responses e.g. kinetics of antibody titers, cell mediated response, etc. The problem of defining optimal schedules is particularly important in cancer immunopreventive approaches, which requires a sequence of vaccine administrations to keep a high level of protective immunity against the constant generation of cancer cells over very long periods, ideally for the entire lifetime of the host.

The Triplex vaccine represents a clear example of such immunopreventive approaches. It has been designed to raise the immune response against the breast cancer for the prevention of the mammary carcinoma formation in HER-2/Neu mouse models using a *Chronic schedule* in a follow up time between 52 and 57 weeks.

However it is not known if the *Chronic schedule* schedule is minimal, i.e. if it can guarantee survival for the mice population avoiding unnecessary/redoundant vaccine administrations. Shorter heuristic protocols failed, in *in vivo* experiments, in fulfilling this requirement, but between the Chronic and the shorter schedules there is still a huge number of possibilities which remain yet unexplored.

The SimTriplex [1] is a *physiological* computational model developed with the aim to answer to this question. It demonstrated able to reproduce *in silico* the *in vivo* Immune System (IS) - breast cancer competition elicted by the Triplex vaccine.

Optimal search strategy was biologically guided. Considering that Chronic proved to be effective for tumor control, the optimal search tried to find a protocol with minimum number of vaccine administrations able to reproduce, in silico, the time evolution of the chronic schedule. This strategy was used in the GA optimal search. It is well known that GA are slowly converging algorithms; the GA optimal search required several days using 32 nodes of an High Performance Computing infrastructure.

To this end we decided to investigate the applicability of **Simulated Annealing** (SA), a global optimization algorithm, widely tested

and known for its computational speed and ability to achieve optimal solutions. (Interested readers can found an extended description of SA algorithm in [8]). The combination of the Simulated Annealing algorithm with biologically driven heuristic strategies, leads to a much faster algorithm and better results for the optimal vaccination schedule problem for Triplex vaccine. In this context we remark how the COMETA grid infrastructure demostrated an excellent framework for protocol search and validation. We first executed the SA algorithm on a subset of the virtual mice population using MPI jobs. We therefore checked the protocol quality calculating (with simple jobs) the survivals of the entire population.

II. The algorithm

The work done by Kirkpatrick [9] opened the path to a deep analogy between Statistical Mechanics (the behavior of systems with many degrees of freedom in thermal equilibrium at a finite temperature) and Combinatorial Optimization (the method of finding the minimum, if any, of a given function with respect to many parameters). There is a close similarity, indeed, between the procedure of *annealing* in solids and the framework required for optimization of complex systems.

A. The optimal vaccination schedule search problem

The SimTriplex model [1] has been created with the aim to mimic the behavior of the immune system stimulated by the Triplex vaccine. It simulates all the major interactions of cells and molecules of the immune system in vaccinated as well as naive HER-2/neu mice. *In silico* experiments showed an excellent agreement with the *in vivo* ones.

As previously said, a protocol is said to be optimal if it can maintain efficacy with a minimum number of vaccine administrations. As in standard drug administration, the vaccination protocol must assure survival for a high percentages of patients. Schedule design is usually achieved using *medical consensus*, i.e. a public statement on a particular aspect of medical knowledge available at the time it was written, and that is generally agreed upon as the evidence-based, state-of-the-art (or state-of-science) knowledge by a representative group of experts in that area. Our goal is to improve medical consensus, helping biologists in design vaccine protocols with simulators and optimization techniques.

Let us consider a time interval $[0, T]$, in which we study the action of the vaccine on a set of virtual mice S. We then discretize the given time interval in $N-1$ equally spaced subintervals of width Δt (=8 hours), i.e. $\{t_1 = 0,\ t_2,\ \ldots, t_i,\ \ldots,\ t_N = T\}$.

Let $\mathbf{x} = \{x_1, x_2, \ldots, x_i, \ldots x_N\}$ be a binary vector representing a vaccine schedule, where $x_i = 0/1$ means respectively administration/no administration of the vaccine at time t_i. The number of vaccine administrations is given by $n = \sum_{i=1}^{N} x_i$. With $T = 400$ days, and $\Delta t = 8$ hours the search space D has cardinality 2^{400}, excluding any exhaustive search.

One of the wet biologists requirements imposes no more than two administrations a week (monday and thursday) because this is already considered a very intensive schedule from an immunological point of view. This reduces the cardinality of the search space D, from 2^{400} ($\sim 10^{120}$) to 2^{114} ($\sim 10^{34}$). We still have no chance for an exhaustive search.

The time of the carcinoma in situ (CIS) formation is computed through SimTriplex simulator. It is defined by $\tau(\mathbf{x}, \lambda_j)$, which is a function of the vaccination schedule $\mathbf{x}$ administered to the mouse $j \in S$ and a parameter λ_j which represents the biological diversity. The vaccine will be obviously effective if $\tau \geq T$.

As pointed out in § 1, any optimal protocol should try to reproduce , *in silico*, the chronic time evolution of cancer cells. This leads us to the need to use two thresholds on the allowed maximum number of cancer cells.

We also note here that, due to biological variability, a schedule found for a single mouse

is not immunologically effective as it usually does not reveal able to protect high percentages of the treated patients. Having this in mind, we formulate our optimization problem as follows.

Let $\{j_1, j_2, \ldots, j_m\} \subset S$, with $m = 8$, a random chosen subset of *in silico* mice, the problem is defined as:

$$\begin{cases} \tau(\bar{\mathbf{x}}, \lambda_{j_1}) & = & \max(\tau(\mathbf{x}, \lambda_{j_1})) \\ \tau(\bar{\mathbf{x}}, \lambda_{j_2}) & = & \max(\tau(\mathbf{x}, \lambda_{j_2})) \\ \vdots & & \vdots \\ \tau(\bar{\mathbf{x}}, \lambda_{j_m}) & = & \max(\tau(\mathbf{x}, \lambda_{j_m})) \\ n(\bar{\mathbf{x}}) & = & \min(n(\mathbf{x})) \\ \text{subject to:} & & \\ M_1(\bar{\mathbf{x}}) & \leq & \gamma_1,\ t \in [0, T_{in}] \\ M_2(\bar{\mathbf{x}}) & \leq & \gamma_2,\ t \in [T_{in}, T] \end{cases} \quad (1)$$

where $M_1(\mathbf{x})$ and $M_2(\mathbf{x})$ are the maximum number of cancer cells in $[0, T_{in}]$ (cellular-mediated controlled phase) and in $[T_{in}, T]$ (humoral-mediated controlled phase) respectively, and $T_{in} \sim T/3$, while γ_1 and γ_2 represent cancer cells threshold in $[0, T_{in}]$ and in $[T_{in}, T]$, respectively.

We deal with a multi-objective discrete and constrained optimization problem.

We modified this last formulation of the problem grouping all the $\tau(\mathbf{x}, \lambda_{j_h})$ $(h = 1, \ldots, m)$ by a proper statistical indicator. We chose the harmonic mean H of the survivals:

$$H(\mathbf{x}, \lambda_{j_1}, \ldots, \lambda_{j_m}) = \frac{m}{\sum_{i=1}^{m} \frac{1}{\tau(\mathbf{x}, \lambda_{j_i})}} \quad (2)$$

since it is very frequently used when statistic measurements of time are involved.

Therefore, the system (1) translates as:

$$\begin{cases} H(\bar{\mathbf{x}}, \vec{\lambda}) & = & \max\left(H(\mathbf{x}, \vec{\lambda})\right) \\ n(\bar{\mathbf{x}}) & = & \min(n(\mathbf{x})) \\ \text{subject to:} & & \\ M_1(\bar{\mathbf{x}}) & \leq & \gamma_1,\ t \in [0, T_{in}] \\ M_2(\bar{\mathbf{x}}) & \leq & \gamma_2,\ t \in [T_{in}, T] \end{cases} \quad (3)$$

with $\vec{\lambda} = (\lambda_{j_1}, \ldots, \lambda_{j_m})$.

B. Simulated annealing

An acclaimed Monte Carlo method, commonly referred as the *Metropolis criterion*, has been designed by Metropolis [10] with the aim to mimic the evolution of the complex systems towards equilibrium at a fixed temperature. This method randomly perturbates the position of the particles of a solid modifying its *configuration*. If the energy difference, ΔE, between the unperturbed and perturbed configurations is negative, the new configuration has lower energy and it's considered as the new one. Otherwise the probability of acceptance of the new configuration is given by the *Boltzmann factor* (which expresses the "probability" of a state with energy E relative to the probability of a state of zero energy).

After a large number of perturbations the probability distribution of the states should approach the Boltzmann distribution.

The Metropolis algorithm can also be used in combinatorial optimization problems to generate sequences of configurations of a system using a *cost function* C and control parameter c respectively as the energy and temperature in the physical annealing.

The SA algorithm can be represented as a sequence of Metropolis algorithms evaluated at a sequence of decreasing values of the control parameter c. A generalization of the method is given as follows: a generation mechanism is defined so that, given a configuration i, another configuration j can be obtained by choosing at random an element in the neighborhood of i.

If $\Delta C_{ij} = C(j) - C(i) \leq 0$, then the probability that the next configuration is j is given by 1; if $\Delta C_{ij} > 0$ the probability is given by $e^{-\Delta C_{ij}/c}$ (Metropolis criterion).

This process is continued until the probability distribution, P, of the configurations approaches the Boltzmann distribution, which translates as:

$$P\{configuration = i\} = \frac{1}{Q(c)} \cdot e^{-C(i)/c} \quad (4)$$

where $Q(c)$ is a normalization constant depending on the control parameter c.

The control parameter is then lowered in steps, with the system being allowed to reach equilibrium by generating a chain of configurations at each step. The algorithm stops for some fixed value of the control parameter c where virtually no deteriorations can be accepted anymore.

At the end the final *frozen* configuration is assumed as a solution of the problem.

C. Implementation

In our *in silico* experiment, we select a population of 200 virtual mice and a simple random sample of $k = 8$ mice. To use the SA algorithm for the optimal vaccine schedule search problem we tried to define the SA relevant concepts (the solid configuration, the temperature, the energy and the semi-equilibrium condition) in terms of vaccine protocol and to describe the main protocol elements (the number of injections, the mean survival age, the time distribution of injections) in terms of a cooling process.

As previously said, we describe any candidate protocol as a binary vector $\mathbf{x}$ of cardinality $V = 114$. The total number of vaccine administrations n and the total number of possible schedules with n vaccine administrations $\mathcal{M}$ are given by:

$$n = \sum_{j=1}^{V} x_n^l(j)$$

$$\mathcal{M} = V!/[n!\ (V-n)!]$$

The configuration distribution is defined by $\mathbf{x}_{in}$, n_{in}, and the initial energy E_{in} as defined later on.

The *temperature* is slowly but constantly lowered to reach a state with minimum energy. We coupled this entity with n, the *number of vaccine administrations* of a semi-equilibrium configuration.

At a given temperature, a semi-equilibrium configuration is reached when its *Energy* is minimal. Since we want to maximize survival times of a mice sample set, the concept of Energy can be easily associated with the *harmonic mean* H of the survival times $\tau_i, H(\bar{\mathbf{x}}, \vec{\lambda})$ (i.e. $E \propto H$). As a matter of fact H decreases when the cumulative survival time of the sample increases, in perfect accord with the energy definition.

The perturbation of a protocol has been initially implemented as random "1" bits reallocation. This perturbation has been heuristically improved using *biological* knowledge. As we want to optimize mice survival, scheduling many vaccine administrations after the death of a mouse makes nonsense. So, in moving from $\mathbf{x}_i^j$ to $\mathbf{x}_i^{j+1}$, we improve random bits reallocation also moving some "1" at a suitable time $t < min\{\tau_i\}\ , i = 1, k$.

The SA algorithm for protocol optimization. *i*) starts from a randomly chosen initial vaccine distribution and finds the initial semi-equilibrium configuration n_{in}, $\mathbf{x}_{nin}^{l_{in}}$, $E_{nin}^{l_{in}}$

ii) Decrease the number of injections of 1 unit; *iii*) find a semi-equilibrium configuration $\mathbf{x}_i$, E_i according to Metropolis algorithm; *iv*) cycle on (*ii*). The algorithm stops when, once the algorithm control parameter, i.e. the number of vaccine administrations, is decreased from n to $n-1$, the Metropolis algorithm is not able to find a semi-equilibrium configuration, i.e. an acceptable value of survivals, in λ iterations. The accepted protocol is the last found at temperature n.

III. Computational results and conclusions

In silico optimal protocol search is a two-step process: *search* and *validation*. During the search step, the optimization tecnique tries to find an optimal protocol. As pointed out in section II-A, optimal search stragies have to be executed on a representative subset of the population in order to guarantee significative survival percentages for the population.

The search technique therefore needs to test simultaneously every candidate protocol on the mice subset to compute its fitness function. This process usually requires a relatively small number of nodes with high communication through-

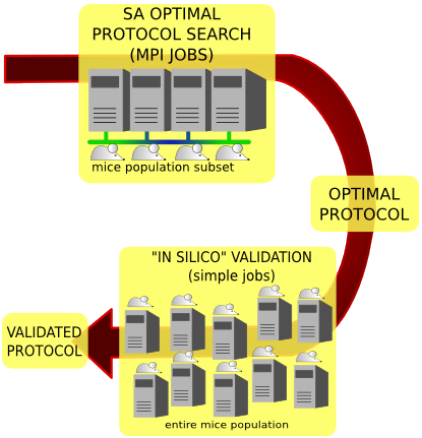


Fig. 2. The "in silico" *search* and *validation* optimal protocol representation.

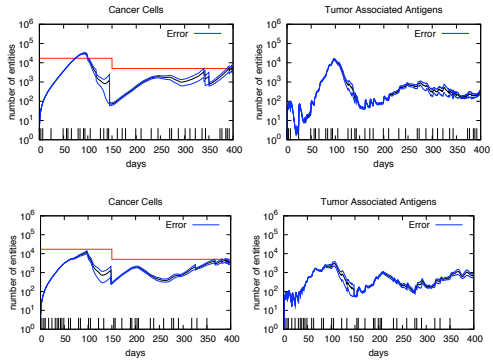


Fig. 3. Cancer cells behaviors and thresholds in GA (top) and SA (bottom). Small vertical bars on the x-axis represent vaccine administration times. Broken-line graphs on the lhs represent safety thresholds.

puts, representing a typical massive parrallel application. In our case it has been implemented using the MPI (Message passing Interface) libraries.

Validation represents a completely different process. Here the protocol found by the search technique is tested over the entire population to compute mean survival rates, requiring an high number of CPUs with almost no need of communication. The "*search and validation*" process is represented in figure 2.

In this context the Cometa grid revealed itself as an excellent tool for our needs. It demonstrated to be highly flexible, giving us the opportunity to execute these so-different processes on the same infrastructure. We only needed to define the first process as an *MPI job* and the second as a sequence of *simple jobs* in a trasparent way, without worrying of the hidden the architectures behind the job submission interface.

To compare the results with those obtained using GA optimization technique [11], we executed the SA algorithm on the same 8 random selected virtual mice sample used by GA. The protocol was then validated on the same population set (200 virtual mice).

The SA *in silico* tumor free percentages of the mice population show no substantial difference with GA results (87% for GA vs $86,5\%$ for SA). Figure 3 shows the mean number of cancer cells, computed on the 200-mice set, for the GA-protocol (up lhs) and the SA-protocol (down lhs). Only the SA-protocol is able to totally fulfill the safety threshold conditions (shown in red).

Moreover the SA algorithm required a computational effort of about 2 hrs on a 8 processor unit to find a protocol with 37 vaccine administrations, showing speed-up factor of $\sim 1.4 \cdot 10^2$ [12] in respect to previous GA experiments.

Conclusions. We presented a new approach to search optimal vaccination schedules. The use of the SA algorithm reaches an *optimal* solution with a highly reduced computational effort.

The SA-protocol vaccine administration distribution is different than the GA-protocol distribution suggesting that many protocols $\mathbf{x_n}$ capable of assuring the same global survival goal may exist.

We additionally found that the grid infrastructure can offer, besides great computational and storage capabilities, excellent flexibility, revealing as a perfect tool for scientists and reseaches.

ACKNOWLEDGMENT

This work was supported under the EC contract FP6-2004-IST- 4, No.028069 (Immuno-

Grid). This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at: http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] Pappalardo, F., Lollini, P.L., Castiglione, F. and Motta, S. Modeling and simulation of cancer immunoprevention vaccine. Bioinformatics. 21, 2891-2897

[2] Neil Kumar, Bart S. Hendriks, Kevin A. Janes, David de Graaf and Douglas A. Lauffenburger Applying computational modeling to drug discovery and development Drug Discovery Today Volume 11, Issues 17-18, September 2006, Pages 806-811

[3] Davies,M.N., Flower,D.R. (2007) Harnessing bioinformatics to discover new vaccines, *Drug Discovery Today*, **12(9-10)**, 389-395.

[4] Kirschner,D.,Panetta,J.C., (1998) Modeling immunotherapy of the tumor-immune interaction. *J. Math. Biol.*, **(37)**, 235-252.

[5] Nani,F.,Freedman,S. (2000) A mathematical model of cancer treatment by immunotherapy, *Math. Biosci.*, **163**, 159-199.

[6] Castiglione,F., Piccoli,B. (2007) Cancer immunotherapy, mathematical modeling and optimal control, *J. Theo. Biol.*, 247 , pagg. 723 - 732.

[7] Agur,Z., Hassin,R., Levy,S. Optimizing chemotherapy scheduling using local search heuristics, Operations Research, 54:5, 829-846.

[8] Van Laarhoven,P.J.M., Aarts,E.H.L. (1987) Simulated Annealing: Theory and Applications, *Springer Edition*.

[9] S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi Optimization by simulated Annealing Science, Vol. 220, No. 4598, 671-680, 1983

[10] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, "Equation of State Calculations by Fast Computing Machines", Journal of Chemical Physics, Vol. 21, pp. 1087-1092, (1953)

[11] Lollini, P.L., Motta, S. and Pappalardo, F. Discovery of cancer vaccination protocols with a genetic algorithm driving an agent based simulator. BMC. Bioinformatics. 7, 352.

[12] Pennisi,M., Catanuto,R., Pappalardo,F., Motta,S. (2008) Optimal vaccination schedules using Simulated Annealing, Bioinformatics, 24:15, 1740-1742.

Marzio Pennisi graduated in Computer Science in March, 2006, at University of Catania, Italy. In December of the same year, he won the competition for Ph.D. course of study at the University of Catania, Italy. He is actually studying models, mathematical methods and computer simulation of the immune system. In particulary he is working on models of mammary tumor growth, diffusion of metastasis and therapy. His present scientific interests range from BioMaths BioComputing and BioInformatics Immunomics, evolutive algorithms, constrained and unconstrained optimization.

Francesco Pappalardo earned his master and PhD. degree from University of Catania, in 2000 and 2004 respectively. Actually he is holding a postdoctoral fellow and he is temporarily assigned to the professorship of Mathematics at the University of Catania. From 2001 to 2004 he did research in computer operating system security. In 2008 he was visiting scholar at Dana-Farber Cancer Institute in Boston (USA) while in 2006 he visited the Molecular Immunogenetics Labs, IMGT in Montpellier (France). His major research areas are on modeling human and mouse immune system responses involved in several pathologies, including viral infections, tumors, and atherosclerosis. This effort leaded and actually leads to both a better understanding of the immune system, and to the application of these modeling approaches to vaccine research. Up to now, Francesco Pappalardo published more than 40 reviewed research papers. He serves scientific community as editorial board member, reviewer and program committee member for major journals and conferences in the area of computer science and computational biology.

Santo Motta was born in Rome in 1946. He obtained "Laurea" in Physics at the University of Catania (1970) and its M.Sc. in Applied Mathematics at the University of London, Queen Mary College (1971). Actually he is associate professor of Mathematical Physics at the Faculty of Pharmacy of the University of Catania and he is member of the Dept. of Maths & Computer Science of the University of Catania. Presently he is interested in computational models of the cancer immune system competition induced by an in vivo tested immune prevention vaccine. Its present scientific interests are BioMaths, BioComputing and BioInformatics, Immunomics. Previously he was interested in numerical methods for transport equation, Non linear waves in classic and relativistic fluid?dynamics, General Relativity and Cosmology, Solar Physics. Up to now, Santo Motta published more than 80 reviewed research papers and he is member of editorial board and reviewer of several international journals.

Grid Based Cancer Growth Simulations

Davide Alemani[1], Francesco Pappalardo[3], Emilio Mastriani[2], Marzio Pennisi[3] and Santo Motta[3]

[1]*EPFL, Lausanne, Switzerland, davide.alemani@epfl.ch*

[2]*Consorzio COMETA, Catania, Italy, mastriani@dmi.unict.it*

[3]*University of Catania, Catania, Italy, {francesco,mpennisi,motta}@dmi.unict.it*

***Abstract*—In this work we introduce a new numerical method to simulate tumor growth and immune system, by applying an hybrid Cellular Automata-Lattice Boltzmann (CA-LB) approach. The CA-LB approach consists in using a Cellular Automata method to keep track of the immune system and the tumor shape and a Lattice Boltzmann diffusion formulation to follow the variation of nutrient concentrations in the microenvironment of the tumor. The main aim is to use the CA-LB model to describe the interactions between a growing tumor next to a nutrient source and the immune system of the host organism. The starting points are two models independently developed. The first model is a reaction-diffusion model for tumor growth in presence of nutrients. It was improved by with the addition of a probabilistic CA model able to keep track of the immune system response to tumor growths and cell to cell adhesion. The second model is based on a detailed description of the immune system at the cellular level with an agent-based method. The model is successfully used for cancer immunoprevention vaccine applications in mice. However, tumor growth is not modeled in detail and diffusive nutrient effects are missing. The COMETA GRID infrastructure was used as a key access point to simulate several cancer growth scenarios that reproduced different cancer growths observed in clinical cases.**

***Index Terms*—Cancer, Agent based models, Hybrid models, GRID**

I. Introduction

In this work we introduce a new numerical method to simulate tumor growth and immune system, by applying an hybrid Cellular Automata Lattice Boltzmann (CA-LB) approach. The CA-LB approach consists in using a Cellular Automata method to keep track of the immune system and the tumor shape and a Lattice Boltzmann diffusion formulation to follow the variation of nutrient concentrations in the micro-environment of the tumor. The main aim is to use the CA-LB model to describe the interactions between a growing tumor next to a nutrient source and the immune system of the host organism.

The starting points are two models independently developed in [1] and [2]. The first model, [1], is a reaction-diffusion model for tumor growth in presence of nutrients. It was improved by [3] and [4] with the addition of a probabilistic CA model able to keep track of the immune system response to tumor growths and cell to cell adhesion. However, the immune system is described only with natural killer (NK) cells and cytolytic T lymphocytes (CTL).

The second model, [2], is based on a detailed description of the immune system at the cellular level with an agent-based method. The model is successfully used for cancer immunoprevention vaccine applications in mice [5]. However, tumor growth is not modeled in detail and diffusive nutrient effects are missing. Both models have shown good agreement in reproducing different morphologies of growing tumors [4] and experimental data in naive and vaccinated mice [2]. The main idea is to combine the capabilities of both these models in order to qualitatively and quantitatively describe the micro-environment of a tumor growth, the diffusion of nutrients and the response of the immune system together with possible immunoprevention vaccine

applications.

THE MODELS

The reaction-diffusion model for nutrients

The nutrient evolution is based on a macroscopic reaction-diffusion model [1], [3]. According to [6], [7], nutrients are often classified into two categories: essential and not essential for cell proliferation. An example of essential nutrient is iron, essential for DNA synthesis and therefore for cell division. Other nutrients affect mainly the motility and death of cancer cells [1]. We will call the former class of nutrients as *proliferative* nutrients and denote their local concentration with $N(\mathbf{x},t)$, while the latter as *survival* nutrients and denote their local concentration with $M(\mathbf{x},t)$.
The proliferative and the survival nutrients satisfy the following reaction-diffusion equations respectively:

$$\frac{\partial N(\mathbf{x},t)}{\partial t} = D_{\mathrm{N}}\nabla^2 N(\mathbf{x},t) - k_{\mathrm{N}}\Big(T+I+\lambda_{\mathrm{N}}C\Big)N(\mathbf{x},t) \quad (1)$$

$$\frac{\partial M(\mathbf{x},t)}{\partial t} = D_{\mathrm{M}}\nabla^2 M(\mathbf{x},t) - k_{\mathrm{M}}\Big(T+I+\lambda_{\mathrm{M}}C\Big)M(\mathbf{x},t) \quad (2)$$

The reaction-diffusion equations (1) and (2) describe the interaction between the nutrients and the cells and the diffusion of the nutrients. The interaction nutrients-cells is described with linear first order reactions.

The probabilistic CA model for tumor growth

The law of the tumor growth are based on probabilistic rules already investigated and successfully used by [3], [1], [8]. In the model we distinguish between cancer (C) and necrotic (D) cells. The cancer cells are able to duplicate, move and die in the tissue, while the necrotic cells are created when a cancer cells dies and do not perform any action i.e. they remain in a necrotic state. They are not consumed by the nutrients. Three parameters regulate the tumor growth: i) θ_{div} tunes the division action, ii) θ_{mov} tunes the propagation action and iii) θ_{del} tunes the death action. Tuning accurately these parameters allows us to reproduce different types of tumor patterns.
The probability laws are the following:

$$P_{\mathrm{div}} = 1 - e^{-\left(\frac{N}{C\theta_{\mathrm{div}}}\right)^2} \quad (3)$$

$$P_{\mathrm{mov}} = 1 - e^{-C\left(\frac{M}{\theta_{\mathrm{mov}}}\right)^2} \quad (4)$$

$$P_{\mathrm{del}} = e^{-\left(\frac{M}{C\theta_{\mathrm{del}}}\right)^2} \quad (5)$$

The rules should taking into account the behavior of a real tumor. They mimic the fact that:

- Cell migration increases near the border of the tumor
- In region of low proliferative nutrient concentration and with a high population of cancer cells, cell division is inhibited and the probability of cell's death increases.
- In region of high survival nutrient concentration and with a high population of cancer cells, cell migration is enhanced and the probability of cell's propagation increases.

The LGCA model for the immune system

The LGCA method for the immune system is based on the SimTriplex simulator [2]. SimTriplex is based on a modification of the Celada-Seiden framework [9].
SimTriplex mimics at the cellular level the behavior of immune cells of vaccinated as well as naive mice. This is a generalized cellular automaton and it is specialized in modeling mammary carcinoma, Triplex vaccine and immune system competition.
In order to simulate immune system competition, the following entities and interactions were added in SimTriplex:

1) the cancer cells (C) which encode their tumor associated antigens (TAA); they interact with Ab, TC and NK.
2) the vaccine cells (VC) which include TAAs, IL-12 and allogenic MHCI; VC

interact like C, with Ab, TC and NK, but the affinity function is modified by the presence of the two adjuvants.

3) the natural killer (NK) cells which do not encode receptors; they interact with Ab, VC and C.

A detailed explanation of the SimTriplex model and the symbols used in the list above is given in [2].

The main difference between the original, see [2], and the modified versions of Simtriplex is in the regulation of tumor growth. The three probabilistic functions, given by equations (3), (4) and (5), were added in the original version of SimTriplex, giving raise to a modified SimTriplex version. Most of the models are naive because they offer a simplistic description of biological systems. Now we have a capacity to link genetic, functional and physical aspects of these models. The key is to determine a more complete sets of probabilistic rules that represent various actual situations and multiple forms of tumors and hallmarks (thing that make tumors what they are).

REALISTIC SIMULATIONS OF TUMOR GROWTH

After validation, we have performed several simulations with the aim to reproduce realistic tumor patterns under different conditions. We have performed qualitative comparison with actual melanoma using the Linux cluster at the Cancer Vaccine Center of the Harvard Medical School in Boston (USA). We ran the numerical algorithm with the aim to reproduce typical tumor patterns from actual melanomas. The values of the parameters were chosen to be as more realistic as possible based on values given in the literature. We have done four different simulations that aim to represent four types of tumors.

The results of each simulation are reproduced in figures 1, 2, 3 and 4.

Figure 1 shows a typical configuration of tumor with uneven edges. We notice that the immune cells align themselves at the border of the tumor mass, with a tendency to accumulate close to the nutrient. Necrotic cells tend to accumulate near the source of nutrient, as well. Most likely, this means that these are sites where cancer cells appear and then instantly die. Indeed, necrotic cells do not consume nutrients, therefore there are no reason that they get close to the source of nutrient (in our case the blood vessel) unless we make the hypothesis that cancer cells migrate close to the source of nutrient attracted by its presence and then die, becoming necrotic cells. The tumor shows a finger-like pattern. It is evident that the presence of the nutrient source is a stress factor for the tumor mass. Tumor cells are attracted by the presence of food, then they die and fractal-like patterns appear. It is interesting to observe that the cancer cells did not exhibit a finger-like pattern but they prefer to accumulate at the border of the tumor mass and close to the source of nutrient, generating a cancer proliferative layer. Indeed, the presence of such a proliferative layer is known in the literature [10], [11], [12].

Figure 2 shows the results of a simulation where the migration probability is much larger than the simulation shown in figure 1. The final distribution shows a tumor with less border irregularity but more compact surface that that shown in figure 1. We notice that necrotic cells are absent. We argue that this kind of tumor are benign, because the necrosis process did not start. Cancer and immune cells show the same tendency as depicted in figure 1. Accumulation of cancer cells near the source of nutrient determines an asymmetric distribution less pronounced than that shown in figure 1, probably due to the absence of necrotic cells. This can be explained considering the different nature of cancer and necrotic cells. Necrotic cells do not perform any action, they are like inert cells, while cancer cells are living cells that are "hungry" of food much more than normal tissue cells. As a consequence, all cancer cells are attracted to the nutrient source. If during this attraction process, some of the cancer cells die and become necrotic cells, a discontinuity appears: where necrotic cells appear, the cancer cells present in their neighbor find themselves

surrounded with more food, thus they slow down their attraction process to the nutrient source creating irregularity in the tumor shape. Comparison of figure 2 with figure 1 shows that the presence of the nutrient plays an important role in determining the shape of a tumor and consequently its benign or malignant nature.

In the simulations shown in figures 3 and

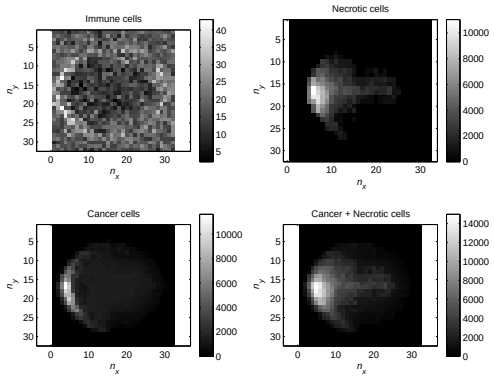


Fig. 1. Simulation A. Parameters are set as follows: $K_n = 10^{-5}, \theta_{div} = 10^{-3}, \theta_{mov} = 10^{5}, \theta_{del} = 10^{-2}$

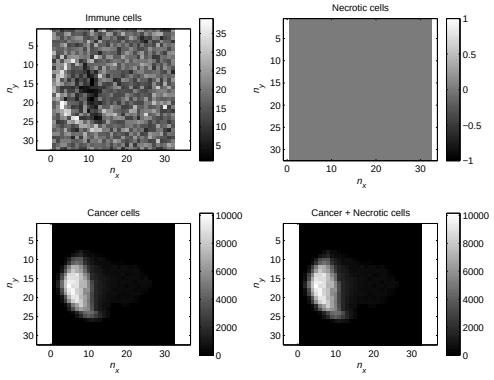


Fig. 2. Simulation B. Parameters are set as follows: $K_n = 10^{-5}, \theta_{div} = 10^{-3}, \theta_{mov} = 10^{3}, \theta_{del} = 10^{-1}$

4 the tumor growth model parameters are the same, but cancer cells are more aggressive in simulation C (the rate constant k_{N} is larger in simulation C than in simulation D). As a result, the tumor shape are similar and they show necrotic cores and border irregularity. The peak number of necrotic cells is about 10^4 in simulation C and about $5 \cdot 10^4$ in simulation D. Among the total number of cancer and necrotic cells, in both simulations 37% are cancer cells and 63% are necrotic. However, although both simulations stop after 50 time steps, the total number of tumor cells (cancer and necrotic) are 10^6 in simulation C and $4 \cdot 10^6$ in simulation D. The less amount of tumor cells in simulation C can be explained by considering the more aggressivity of cancer cells in consuming nutrients. Cancer cells in simulation C consume nutrient more rapidly than in simulation D, therefore, after a same amount a time, in some regions there will be less nutrient available for consumption in simulation C than in simulation D. This fact determines a decrease of the division probability of cancer cells P_{div} in simulation C and this may explain why the total number of tumor cells is lower in simulation C (figure 3) than that in simulation D (figure 4).

In figure 5 we show all the four simulations A,

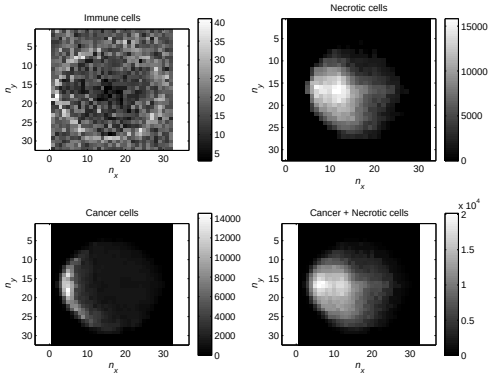


Fig. 3. Simulation C. Parameters are set as follows: $K_n = 10^{-6}, \theta_{div} = 10^{-3}, \theta_{mov} = 10^{5}, \theta_{del} = 10^{-2}$

B, C and D and a possible link with four different types of real tumor found in the literature.

CONCLUSIONS

The main conclusion of our work reproduced in this paper, can be summarized as follows:

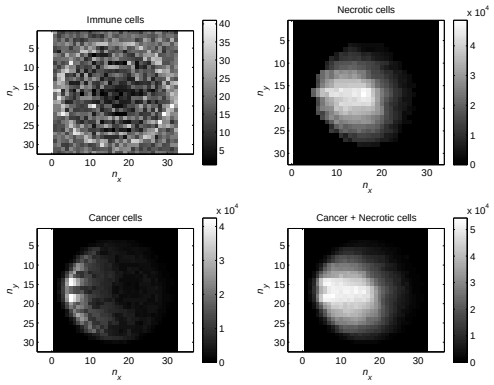


Fig. 4. Simulation D. Parameters are set as follows: $K_n = 10^{-7}, \theta_{div} = 10^{-3}, \theta_{mov} = 10^{5}, \theta_{del} = 10^{-2}$

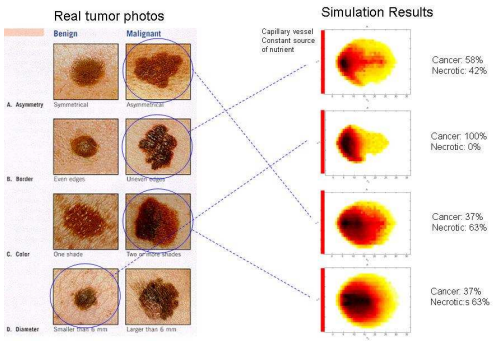


Fig. 5. The results from the four simulations A, B, C and D are related with four different real tumor shapes taken from internet at the following website: http://www.southcoastmedcenter.com/graphics-/lab/abcd_melanoma.jpg. A red-yellow color scale has been used to better match the real tumor shapes.

- Our model is able to reproduce early stage avascular tumors. In particular, it is able to correctly predict the formation of three main stages: i) the necrotic cores made with necrotic cells, ii) the quiescent layer made essentially with cancer and necrotic cells and iii) the proliferating layer made with cancer cells.
- Our model is able to qualitatively reproduce finger-like tumor patterns. Both necrotic and cancer cells show this type of shape. By tuning the parameter that defines the probability of necrosis, it is possible to control the number of necrotic cells. In particular, when the necrosis probability is reduced, we do not observe the formation of necrotic cells, but only cancer cells that accumulate near the source of nutrient.
- Our model is able to qualitatively reproduce tumor shapes with border irregularity, by decreasing the consumption rate of nutrient by the cells. It is interesting to see that more than one necrotic core appear, that is an indication of malignancy in melanomas.

ACKNOWLEDGMENT

This work was supported under the EC contract FP6-2004-IST- 4, No.028069 (ImmunoGrid). This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at: http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] S.C. Ferreira Jr, M.L. Martins, and M.J. Vilela. Reaction-diffusion model for the growth of avascular tumor. *Physical Review E*, 65:021907, 2002.

[2] F. Pappalardo, P.-L. Lollini, Castiglione F., and S. Motta. Modeling and simulation of cancer immunoprevention vaccine. *Bioinformatics*, 21(12):2891–2897, 2005.

[3] D.G. Mallet and L.G. De Pillis. A cellular automata model for tumor-immune system interactions. *Journal of Theoretical Biology*, 239(3):334–350, 2006.

[4] L.G. De Pillis, D.G. Mallet, and A.E. Radunskaya. Spatial tumor-immune modeling. *Computational and Mathematical Methods in Medecine*, 7(2):159–176, 2007.

[5] S. Motta, F. Castiglione, P. Lollini, and F. Pappalardo. Modelling vaccination schedules for a cancer immunoprevention vaccine. *Immunome Research*, 1:5, 2005.

[6] M. Scalerandi, A. Romano, G.P. Pescarmona, P.P. Delsanto, and C.A. Condat. Nutrient competition as a determinant for cancer growth. *Physical Review E*, 59(2), 1999.

[7] M. Scalerandi, G.P. Pescarmona, P.P. Delsanto, and B. Capogrosso Sansone. Local interaction simulation approach for the response of the vascular system to metabolic changes of cell behavior. *Physical Review E*, 63:011901, 2000.
[8] S.C. Ferreira Jr, M.L. Martins, and M.J. Vilela. Morphology transitions induced by chemotherapy in carcinomas in situ. *Physical Review E*, 67.
[9] F. Celada and P. Seiden. A Computer Model of Cellular Interactions in the Immune System. *Immunology Today*, 13(2):56–62, 1992.
[10] N. Bellomo and L. Preziosi. Modelling and Mathematical Problems Related to Tumor Evolution and Its Interaction with the Immune System. *Mathematical and Computer Modelling*, 32:413–452, 2000.
[11] N. Bellomo, A. Bellouquid, and E. De Angelis. The Modelling of the Immune Competition by Generalized Kinetic (Boltzmann) Models: Review and Research Perspectives. *Mathematical and Computer Modelling*, 37:65–86, 2003.
[12] N. Bellomo, E. De Angelis, and L. Preziosi. Multiscale Modeling and Mathematical Problems Related to Tumor Evolution and Medical Therapy. *Jounal of Theoretical Medecine*, 5(2):111–136, 2003.

Francesco Pappalardo earned his master and PhD. degree from University of Catania, in 2000 and 2004 respectively. Actually he is holding a postdoctoral fellow and he is temporarily assigned to the professorship of Mathematics at the University of Catania. From 2001 to 2004 he did research in computer operating system security. In 2008 he was visiting scholar at Dana-Farber Cancer Institute in Boston (USA) while in 2006 he visited the Molecular Immunogenetics Labs, IMGT in Montpellier (France). His major research areas are on modeling human and mouse immune system responses involved in several pathologies, including viral infections, tumors, and atherosclerosis. This effort leaded and actually leads to both a better understanding of the immune system, and to the application of these modeling approaches to vaccine research. Up to now, Francesco Pappalardo published more than 40 reviewed research papers. He serves the scientific community as editorial board member, reviewer and program committee member for major journals and conferences in the area of computer science and computational biology.

Davide Alemani graduated from the University of Milan, Italy in 2001 with a Masters degree in physics. His thesis was based on the theory and applications of classical and modern mathematical methods to solve partial differential equations in applied sciences. He continued his career at the Department of Chemistry of the University of Geneva obtaining a PhD in computational science for environmental chemistry in 2007. He worked on the application of the Lattice Boltzmann Method to compute metal fluxes in environmental systems. During this time he successfully developed a numerical code (MHEDYN) which is being adopted by the Chemistry community. In 2008, he was granted a scholarship by the Swiss National Foundation in which he spent 12 months at the University of Queensland, Australia, working on a research project (IMMUNOGRID) in computational modeling for the tumor growth and immune systems. He is currently pursuing a post-doctorate in the field of molecular dynamics at the EPFL in Lausanne, Switzerland.

Marzio Pennisi graduated in Computer Science in March, 2006, at University of Catania, Italy. In December of the same year, he won the competition for Ph.D. course of study at the University of Catania, Italy. He is actually studying models, mathematical methods and computer simulation of the immune system. In particulary he is working on models of mammary tumor growth, diffusion of metastasis and therapy. His present scientific interests range from BioMaths BioComputing and BioInformatics Immunomics, evolutive algorithms, constrained and unconstrained optimization.

Santo Motta was born in Rome in 1946. He obtained "Laurea" in Physics at the University of Catania (1970) and its M.Sc. in Applied Mathematics at the University of London, Queen Mary College (1971). Actually he is associate professor of Mathematical Physics at the Faculty of Pharmacy of the University of Catania and he is member of the Dept. of Maths & Computer Science of the University of Catania. Presently he is interested in computational models of the cancer immune system competition induced by an in vivo tested immune prevention vaccine. Its present scientific interests are BioMaths, BioComputing and BioInformatics, Immunomics. Previously he was interested in numerical methods for transport equation, Non linear waves in classic and relativistic fluid?dynamics, General Relativity and Cosmology, Solar Physics. Up to now, Santo Motta published more than 80 reviewed research papers and he is member of editorial board and reviewer of several international journals.

Parameters Estimate in Metabolic Networks Reconstruction

G. Aprea[1], G. Licciardello[2], and V. Rosato[3]
[1]*ENEA, Via del Vecchio Macello, 80055 Portici (Naples), Italy, giuseppe.aprea@gmail.com*
[2]*Science and Technology Park of Sicily, stradale V. Lancia 57, z.i. Blocco Palma I, 95121 Catania, Italy, gralicci@unict.it*
[3]*ENEA, Computinig and Modelling Unit, Via Anguillarese 301, 00123 S.Maria di Galeria (Rome), Italy, rosato@casaccia.enea.it*

***Abstract*—In this paper we describe the implementation of two different parallel codes for parameters estimate in metabolic networks based on the genetic algorithm to fully take advantage of modern computational facilities such as the Enea GRID.**

***Index Terms*—COMETA, journal, LaTeX, paper, template.**

I. Introduction

Advances in omics sciences produce large amounts of data that need analysis and interpretation. Reliable explanations of how processes are regulated require an accurate modeling approach at the systems level. In this paper we focus on metabolic networks models which reproduce the time evolution of all the metabolites. Quite often these models rely on several unknown parameters which have to be estimated from experimental data. This task consists in the solution of an inverse problem which requires the use of an efficient optimization algorithm. GA [1] is a widely known optimum search method which yields reliable values for model parameters with a large computational demand. Our aim is to develop a parallel implementation for parameter estimate based on GA to fully take advantage of the modern different computational facilities. Our implementation relies on:

- Ecell software from Keio University(Japan) [2], [4] for simulations of biochemical networks;
- LSF - load sharing facility - a job scheduler for (multi)cluster [3].

II. The genetic algorithm: Basics

According to GA analogy, in a biochemical network, a set $K = ki, i = 1, m$ of unknown parameters is defined genome. Each of these missing array of constants gives rise to a different behavior of the network, that is different time evolutions for the metabolites' concentrations. The network with the genome end its behavior together constitute an individual and a group of individuals is a population. As in the case of populations of organisms in nature, GA populations undergo a selection where good experimental data fitting represent the selection criterion. After every selection stage, a new population is created; the current generation is over and a new one is ready. After a large number of generations, GA is expected to yield the individuals which best fit experimental data.

III. The genetic algorithm: Computational scheme

GA is implemented following this steps:

1) a random number I of genomes is chosen, each of them differing from the others by the value of the unknown m constants. This is the starting population.

2) for each individual, the time evolution of the metabolites' concentration is calculated with the E-Cell tool.
3) metabolites' concentration allows to evaluate, for each individual, the corresponding value of the cost function: $F_k = \sqrt{\sum [x_i(t_{j+1}) - x_i(t_j)]^2}, (k = 1, I)$; a good fit to data implies a small F_k and viceversa.
4) Mating is performed selecting the individuals of the current generation according to the cost function F_k, the smaller the more likely to be selected. Mutation is also added to yield the final offspring. After mating and mutation, a new generation is ready and and the cycle restarts from point (2) unless we meet one of the following conditions (in these cases, the procedure ends):
   - the best value for the cost function becomes smaller than a fixed threshold;
   - a pre-defined limit number of generations has been reached.

IV. GA AND SPECIFIC COMPUTATIONAL ARCHITECTURES

The efficiency of the GA implementation stems from the choice of a number of parameters: the number of individuals in a population, the number of generations and the specific values of the parameters used for selection, mating and mutation processes. Another important element to save overall time execution is parallelizing the code; for example the computation of the fitness of individuals belonging to the same generation. In this work we suggest efficient implementations of GA, in relation to specific computing architectures: a single cluster architecture (SCA), characterized by many computational nodes tightly interconnected through a low-latency, high-bandwidth network and a GRID (or multicluster architecture), characterized by a number of SCAs linked through a Wide Area Network. The computational problem has thus been mapped on a specific computing architecture by assuming different implementations of the method.

A. SCA implementation

In this case we propose a single-stage parallelism approach. A master node generates an instance of a population of N individuals and then, if γ is the number of computing nodes, allots to each of them the same number N/γ of fitness evaluations. When the fitness of all the genomes has been evaluated, the master node gathers all data from the other computing nodes and performs selection, mating and mutation to provide new offspring which is subsequently reallotted, as before, to the computing nodes. The procedure is repeated for n_G generations.

B. GRID implementation

In the GRID case, the specific computing architecture suggests a different implementation of the method. In this case, in fact, the single larger generation of SCA is replaced by a number γ of smaller "secondary generations" (SG), each composed of N/β genomes. The master node allots the SGs to other different secondary master nodes which accomplish the procedure described above for SCA. This procedure can be seen as the definition of many "islands" where subsets of a generation are allowed to evolve. Each "island" evolves new generations, independently, on the basis of their initial genomes. After a number of generations n_G , the islands are allowed to exchange their resulting genomes. The master node gathers all data from the SGs, ranks their best genomes and send back those with the lowest cost function. Then each SG generates a new seed-population by using half of the best-fitting genomes received from the master and half by randomly generating new individuals. This procedure is iterated n_S times, so that the total number of SG generations is $N_T = n_G n_S$. In this way two main results are achieved: the first is to allow cooperation among SGs, which can exchange best individuals, and the second is keeping diversity to avoid being trapped by local minima effects.

V. Details of the models used for computations

We have considered different networks whose topology and kinetic constants are known. We have generated the time-course of a number of reaction products which are then used as they were experimental results. We have thus "hidden" some of the kinetic constants and asked the method to recover their values. As for the networks chosen, the first describes the cyrcadian rhythm in Drosophila [5] while the second is related to the PHA (polyhydroxyalkanoate) production in Pseudomonas. The two cases have been chosen as paradigmatic of different behaviors: in the first case, the method should be able to recover the frequency and amplitudes of experimental observations; in the second, the large initial production rate and the subsequent time-scale for reaching the asymptotic flat behavior. We briefly report the main features of the biochemical models used for testing the method in Table I, together with the choices made in the different implementations. Before hiding the kinetic parameters, we used their "true" values to obtain the time course of some metabolites: three different metabolites in the case PHA metabolism model and two ones in the case of cyrcadian rhythm. In Tables II and III, we resume the values of the parameters which define the method's implementations on the different computing architectures for the cyrcadian rhythm and the PHA metabolism.

VI. Results and discussion

In this section we report the results obtained with the two model's implementations, for the two metabolic networks. In Table [**?**] we report the numerical results of our GA implementations. Results allow to highlight the following points:

- Both the SCA and the GRID implementations allow to estimate numerical values for the kinetic constants very close to the "true" ones which can reproduce the "experimental" data with a very good approximation. The k_3^{pha} parameter is the only one for which both approaches produce different results and different from the "true" one. A further investigation has shown the existence of a (almost) flat optimum for this parameter i.e. there is a substantial independence of the resulting solutions from the absolute value of this parameter.
- if compared at equal number of genomes processed, the SCA implementation showed a better performance than the GRID one, either in minimizing the cost function and in the number of genomes needed to reach the minimum. This is not much surprising because it is known that GA works better with larger populations;
- the amplitude of maximum fluctuation of the cost function away from the minimum of the best genome of each generation in the SCA approach is about $1.5 - 2$ times larger than that of the best genome from the set of parallel generations of the GRID implementation. This also follows from the ability of a larger population to efficiently span the parameters space.

VII. Conclusion

In this work, we have attempted to show that GA is a powerful tool providing good guesses of unknown kinetic parameters in metabolic networks. It has been used in combination with the E-Cell tool which yields the evaluation of the cost function. Although it is computationally expensive, its workflow is prone to be parallelized on multi-processor computing architectures. We have

- provided evidences on the effective ability to retrieve correct guesses when used to discover unknown kinetic constants in metabolic networks; evidences have been provided for different cases, where the time behavior of the products in the networks varies from exponential decay, as in the case of PHA metabolism in Pseudomonas to an oscillating behavior, as in the case of the cyrcadian rhythm in Drosophila;

- designed two different parallel implementations of the method, to be used on different classes of computing architectures: large homogeneous clusters, with fast interconnection (SCA), and GRID architectures, characterized by loosely coupled computing elements.

The SCA implementation is based on the fast internode's communication which allows a rapid and efficient flux of information between a master node and the computation nodes. The GRID implementation, in turn, introduces a double level of parallelism: each computational node provides the evolution of a subset of the whole population as the loose interconnection does not allow to make every node communicate with a single master at the end of each generation. Only at given frequency, the different population subsets exchange data. We have tested both SCA and GRID implementations to deal with two different cases: in the first, we wanted to reproduce a periodic time behavior of metabolites concentration while in the second we faced a typical slowly-decaying behavior, i.e. an initial increase followed by a slow saturation. In both cases the parallel procedures were able to provide fully satisfactory results both from a qualitative and, more importantly, from a quantitative points of view. The SCA implementation, using a single, large population, is faster than the GRID one in reaching its optimum, even in the case where we compare the performance of the latter with a SCA implementation with a smaller population.

We are working on a further implementation of the method which mixes up features of both the SCA and the GRID approach. In this approach, we will force successive generations to reaching higher population densities in the parameter space, by allowing them to explore just a subset of the whole parameter space. At programmable intervals of generations, a number of subsets with the worst results is discarded and further generations are constrained to make a fine-grain exploration in the region(s) of the parameters space where better solutions have been obtained. We expect this approach to be better in approaching the correct solution and to overcome problems of instability due to the possible presence of more than one stable minimum which plague the SCA approach. Moreover, we will apply this method on further cases of some technological relevance. The proposed method can be run in conjunction with the methods allowing the network to be designed, in order to ultimately provide reliable computational models of metabolic pathways to be used for metabolomic studies and to provide in-silico derived insights for bacterial metabolic engineering [6].

ACKNOWLEDGMENT

This work has been performed in the frame of the project CRESCO (Computational Research Center for Complex Systems) co-founded by ENEA and the Italian Ministry of University and Research in the frame of "Programma Operativo Nazionale 2000-2006 Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione, Misura II.2 : Societ della Informazione per il Sistema Scientifico Meridionale, Azione a : Sistemi di calcolo e simulazione ad alte prestazioni". The authors wish to acknowledge discussions and useful suggestions by I. Arisi (EBRI, Rome) and V.Catara (University of Catania). They would also thank the ENEA colleagues of the ENEA Research Center in Portici for supporting the technical implementation of the work on the high-performance computing architectures of the CRESCO Project.

REFERENCES

[1] D.E. Goldberg, *Genetic Alghorithms in Search, Optimization and Machine Learning*, Boston (MA) USA: Addison-Wesley Longman Publishing Company, 1989.
[2] E-cell [http://www.e-cell.org/ecell].
[3] Platform Computing, [http://www.platform. com].
[4] M. Tomita, K. Hashimoto, K. Takahashi, T. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, J. Venter and C. Hutchison, Bioinformatics **15**, 7284, 1999.
[5] J. Tyson,C. Hong,C. Thron,B. Novak, Biophys J., , **77**(5), 2411, 1999.
[6] DREAM project [http://wiki.c2b2.columbia.edu/dream/index.php/The_DREAM_Project].

Dr. Giuseppe Aprea Graduated in Physics in 2002, finished Ph.D. in Physics in 2006. In 2007 he won a research grant to work at ENEA (Italian National Agency for New Technologies, Energy and Environment) in System Biology project.

Dr. Grazia Licciardello Graduated in Biology in 2002, finished Ph.D. in Technologies for Plants Health in 2008. Since 2003 she has collaborated with the Science and Technologies for Plants Health Department at the University of Catania (DISTEF) in the field of Biotechnologies for Plants Health.

Dr. Vittorio Rosato Graduated in Physics in 1979, finished Ph.D. in Physics in 1986. Since 1987 he has collaborated with ENEA (Italian National Agency for New Technologies, Energy and Environment) becoming confirmed researcher in 1990. He has developed expertise in Statistical Mechanics, Condensed Matter Physics, High Performance computing, Optimization Techniques, Complex Systems Modeling.

TABLE I
MAIN DATA OF THE TWO MODELS USED FOR COMPUTATIONS. LAST COLUMN REPORTS THE NUMBER OF "EXPERIMENTAL" POINTS USED TO EVALUATE THE COST FUNCTION (THE VALUE n_p IN EQ.2).

Model	Metabolites	Enzymes	Processes	$M-m$	m	N. of "experimental" points [Metabolite list]
Cyrcadian rhythm	2	0	5	9	2	10^3[M,P]
PHA metabolism	8	5	5	21	3	9940 [Fatty acid, 2-trans-enoyl-CoA,PHA]

Model	Parameter [Solution]	Solutions from the different approaches			
		SCA	GRID ($N/\beta = 20$)	GRID ($N/\beta = 40$)	GRID ($N/\beta = 60$)
Pha metabolism	$k_1^{pha}[10.0sec^{-1}]$	9.9998	9.6165	10.0118	10.0000
	$k_2^{pha}[1.21e-05M]$	$1.1986e-05$	$1.2494e-05$	$1.2100e-05$	$1.2099e-05$
	$k_3^{pha}[7.2e-04M]$	$27.1001e-04$	$16.1585e-04$	$7.1986e-04$	$6.4814e-04$
Cyrcadian rhythm	$k_1^{period}[200M^{-1}]$	199.9998	199.9978	199.9888	200.0003
	$k_2^{period}[10sec^{-1}]$	10.0000	10.0000	10.0001	9.99999

TABLE IV
NUMERICAL RESULTS FOR THE HIDDEN PARAMETERS. IN THE SECOND COLUMN, WITHIN BRACKETS, WE REPORT THE "TRUE" VALUE OF THE PARAMETER.

Model	n_G	N	γ
cyrcadian rhythm	1000	100	$\sim$100
PHA metabolism	1000	100	$\sim$100

TABLE II
IMPLEMENTATION PARAMETERS FOR THE SCA ARCHITECTURES. n_G THE TOTAL NUMBER OF GENERATIONS THE MODEL HAS BEEN ALLOWED TO GENERATE, N THE NUMBER OF $genome$S IN EACH GENERATION, γ THE NUMBER OF COMPUTING PROCESSORS USED FOR COMPUTATIONS.

Model	N/β	β	n_G	n_S	γ
Cyrcadian rhythm	20	5	200	4	$\sim$10
	40	5	200	4	$\sim$10
	60	5	200	4	$\sim$10
PHA metabolism	20	5	200	4	$\sim$10

TABLE III
IMPLEMENTATION PARAMETERS FOR THE GRID ARCHITECTURES. N/β THE NUMBER OF INDIVIDUALS IN EACH GROUP; β IS THE NUMBER OF GROUPS OF INDIVIDUALS WHICH EVOLVE INDEPENDENTLY; n_G THE NUMBER OF GENERATIONS CREATED IN EACH GROUP BEFORE THE DATA EXCHANGING BETWEEN GROUPS; n_S THE NUMBER OF TIMES THE DIFFERENT GROUPS EXCHANGE DATA (SUCH AS THE TOTAL NUMBER OF GENERATIONS $N_T = n_G \cdot [n_S + 1]$); γ THE NUMBER OF COMPUTING PROCESSORS USED FOR COMPUTATIONS.

PseudoBioRes: A New Comprehensive Database for *Pseudomonas*

Grazia Licciardello[1,2], Vittoria Catara[2], Anshuma Mangtani[3], Rocco Casilli[4] and Vittorio Rosato[3]

[1]*Science and Technology Park of Sicily, Italy, g.licciardello@unict.it*
[2]*Dipartimento di Scienze e Tecnologie Fitosanitarie, Università di Catania, Italy, vcatara@unict.it*
[3]*ENEA, Casaccia Research Centre, Italy*
[4] *Ylichron S.r.l. Roma, Italy, rosato@casaccia.enea.it*

Abstract—***Pseudomonas*-related data are dispersed among many bioinformatics resources and data acquisition of species which genomes have not been sequenced yet represents the most important limit for researchers. Data search and extraction could be possible only via cross researches through different web sources. To overcome this issue we projected a database, designated "PseudoBioRes", that aims to provide an integrated resource collecting information on genes and/or proteins on the basis of their potential applications. The database has been assembled by linking the data to their original sources. As a model, in this first version, a section dedicated to PHA, has been developed. We collected data for 6 specific enzymes involved in the PHA metabolic pathway for 15 species studied, providing a total of 75 gene accessions and 276 protein sequences as well as the genomic contest for those species whose genomes have been completely sequenced.**

Thanks to a user friendly interface the user can browse PHA gene or protein sequences among all *Pseudomonas* spp. and perform multi-gene/protein comparisons.

The database is open source in order to maintain consistency with the new findings and can also be used as a guideline in order to create other sections for other relevant metabolites.

In the next future, thanks to the storage resources and computing capability of the Grid, we aim to improve the data analysis possibilities sharing them on the web with other laboratories. At the moment, it is accessible at the URL: www.ylichron.it/PHA_pseudomonas_DB.

***Index Te1ms* — Pseudomonas, polyhydroxyalkanotes, database.**

I. INTRODUCTION

Pseudomonas Migula 1894, includes bacterial species of relevant interest in medicine, plant pathology and biotechnology, as confirmed by the 35 genome projects on 10 different *Pseudomonas* spp. submitted and the large number (19) in progress or in draft assembly status. For the most important species, e.g. *P. aeruginosa*, *P. putida*, *P. fluorescens* and *P. syringae*, the genome of more than one strain has been sequenced. In the last few years, a huge amount of data has been generated, and more and more are expected in the next years.

This offers an unprecedented opportunity to use the comparative analysis approach in studies of evolution and functional genomics, shedding light on molecular mechanisms regulating different metabolic pathways. In this context, the problem of the optimal extraction of representative datasets of genomic and proteomic data assumes a crucial importance. Genome annotations are accessible directly

from the GenBank (http://ncbi.nlm.nih.gov) and from specialized web sites. The Pseudomonas Genome Database v2 (PGDv2) is a *P. aeruginosa* specialized database in which the genome annotation is continually updated and the database content and functionality. Since 2005, this database provides also annotations of other *Pseudomonas* genomes, and acts as a valuable comparative resource.

Data about *P. syringae*, a plant-pathogenic bacterium, strains of which are characterised for their diverse plant-specific interactions, are collected in the Pseudomonas - Plant Interaction web site (http://pseudomonas- syringae.org/home.html).

Recently, an integrated bioinformatics platform for a *Pseudomonas* systems biology approach to infection and biotechnology has been established. The database called SYSTOMONAS (SYSTems biology of pseudOMONAS) accessible at http://www.systomonas.de encourages the *Pseudomonas* community to elucidate cellular processes of interest [1].

On the other hand, little information is available on strains, which genomes have not been sequenced yet. *Pseudomonas*-related data (gene and protein sequences and metabolic pathways) despite being available for a large number of strains, are, in fact, dispersed among many sources. Information extraction could be accessed only via cross researches from either different web sources dedicated to specific class of enzymes or bacterial species or the GenBank Database.

We faced this problem studying polyhydroyalkanoates (PHA) production by *Pseudomonas* species. Most of the bacteria in this genus are able to produce granules of medium-chain-length poly (3-hydroxyalkanoates) (mcl-PHAs) as energy storage compounds [2]. Once extracted from cells these molecules reveal similar properties to those of common plastic, moreover they are degraded by microbial depolymerases. mcl-PHA genetic locus in *Pseudomonas* spp. [2] consisted of two PHA syntaxes (PhaC1 and PhaC2) [3] separated by the intracellular PHA depolymerase (PhaZ) essential for polymer utilization and biodegradability [4]; a proposed structural protein belonging to the TetR family regulators (PhaD); and two PHA granule-associated proteins (PhaF and PhaI) [5, 6]. Integration of *P. putida* KT2442 classical experimental data along with genomic and high-throughput data stimulated the reconstruction of three different metabolic models aimed to improve PHA production, as a demonstration of the interest of the bioinformatics' community for this metabolic pathway [7, 8, 9].

PHA genes of many *Pseudomonas* strains in addition to those derived from the genome sequencing projects are considerable and dispersed in many database and bioinformatic resources.

Since now, there is no instrument enabling a simple and rapid extraction of *Pseudomonas* related data in a sole comprehensive database.

In the following sections we describe the construction and content of PseudoBioRes, a database which aims to partially fill this void, its graphical interface and usefulness [10].

1) Construction and content

PseudoBioRes aims to generate a specialized *Pseudomonas* resource to complement the available databases in their biological utility and application, providing a comprehensive information of *Pseudomonas*-related sequences and data on gene and protein sequences worldwide available clustered on the basis of the metabolic pathway in which they are involved. In its current release, all the proteins involved in the metabolism of PHA isolated or deduced by the genome sequencing projects in species belonging to the *Pseudomonas* genus, were collected. The proteins clustered on the basis of their sequence similarity and class (PhaC1, PhaC2, PhaZ, PhaD, PhaI, PhaF, PhaG and PhaJ) were also interconnected with genomic data when available. We collected data for 6 specific enzymes involved in the PHA metabolic pathway for 15 species studied, providing a total of 75 gene accessions and 276

protein sequences as well as the genomic contest for species completely sequenced. To complete this section it was necessary to articulate query terms and to manually implement data results for each single species. We collected also the sequence data of 15 genomes.

The database consolidated information from external sources and manually annotated them into a relational database. A search engine tool that allows the query/retrieval of a class of protein in all the *Pseudomonas* species in which it has been sequenced, will be developed. Protein and gene sequences could be extracted and exported simultaneously for all the *Pseudomonas* species ready to be used for *in silico* analysis. The way it provides for the retrieval and extraction of sequences will allows the user to overcome obstacle encountered in the integrative of different bioinformatic resources.

2) Data sources

PseudoBioRes is a result of experimental data provided by different research groups or retrieved by external sources. Among them are Pseudomonas Genome Database v2 (PGDv2, http://www.pseudomonas.com/), GenBank (http://www.ncbi.nlm.nih.gov/), KEGG Kyoto Encyclopedia of Genes and Genomes (http://www.genome.jp/kegg/) and the List of Prokaryotic Names with Standing in Nomenclature–LPSN (http://www.bacterio.cict.fr/).

Data on genome sequences were extracted from the section Gbrowse of PGDv2, which stores and integrates data extracted from the project Pseudomonas Genome Project and from PseudoCAP (Pseudomonas aeruginosa Community Annotation Project). The GenBank was used as gene and protein data source using the engine of NCBI (National Centre for Biotechnology Information).

The PHA database dedicated section was completed with articulated query terms and manually implemented data results for each single species. It took a long time but all data were included

For metabolic pathways and enzyme classes we used the Japanese GenomeNet service, KEGG, which integrates metabolic pathways (data on metabolic pathway and complex), genes (data on functional genes and their protein products) and ligands (Chemical compounds, drugs, glycans, and reactions). From here we extrapolated *Pseudomonas* PHA metabolic pathway.

The occurrence of many DNA sequences obtained from "unknown" strains without any further characterization pointed out a gap between environmental studies and *Pseudomonas* taxonomy. Thus we provided a list and the link of *Pseudomonas* species as retrieved from the LPSN, which includes the nomenclature of prokaryotes and their changes as cited in the Approved Lists of Bacterial Names or published in the International Journal of Systematic Bacteriology (IJSB) or later in the International Journal of Systematic and Evolutionary Microbiology (IJSEM). Genes not attributed to a species were referred as *Pseudomonas* spp. and the strain name was reported as in the NCBI taxonomy database.

3) Structure of the database

PseudoBioRes database has a tree-structure with an introduction page which reports the main goals of database. The content of the database is built on three main interconnected blocks dealing with species, genomes and genes.

From the section "Species" the user is sent to the web pages corresponding to alphabetical list of the 175 species of *Pseudomonas*, taken from LPSN, with a related link to the correct nomenclature, where a particular species can be selected to get its relevant data. Provided links allows to reach Genome sequencing projects (complete and in progress) from the Gbrowser of the *Pseudomonas* genome database v2 web site. A complete comprehensive list is provided from the website www.pseudomonas.com,

showing the ongoing and the completed genome projects related to sequencing of genes of various *Pseudomonas* species.

Starting from a page dedicated to a particular *Pseudomonas* species page it is possible to access 5 fields: general description, NCBI Taxonomy browser, relevant papers, genome sequence (if available) and genes and proteins involved in a specific pathway (in this version only PHA).

The general description has been prepared by using various literature sources focusing upon some general characteristics of the species and PHA production. It reports a brief description and its biological relevance and role in different fields such as clinical, agricultural, environmental. The link to the Tax-browser of NCBI gives more information on taxonomy. It also links to other databases like Genes, Proteins, Genome, Nucleotide, Genome Projects, Structure etc. The relevant scientific literature used to compile the text pages and related to the particular species is given into the "relevant papers" page.

The link to "complete genome" corresponds to GBrowse tool which shows the complete map along with the positioning of genes. It is possible to find here the exact location of specific genes in the genome map for some species using the search engine of that site. In future we plan to replace this link with a better source.

The link to "PHA related genes and proteins" shows the various genes and proteins involved in *Pseudomonas* PHA production. From there, it is possible to get into further pages where genes and proteins related to PHA biosynthesis were collected, into two sets of data. The set "PHA related genes" contains all the genes derived from *Pseudomonas* genome sequence project, when available. From this page it is possible to gain the corresponding page of the Entrez Gene ID of the NCBI web site which provide the genomic context, genomic region, the transcript and product and link to other database (Conserved domain, PubMed, KEGG, taxonomy, TIGR, etc). It allows to directly download the nucleotide sequence in FASTA format and to have information about the metabolic pathway in which the gene is involved thanks to KEGG Database link.

The set "PHA related proteins" contains all the protein sequences derived either from the sequence genome contest or directly from cloned genes. In this case, sequence information was extracted after gene isolation and sequencing and related to PHA yield data. Also in this case it is possible to download the FASTA protein sequence format and have information about the metabolic pathway.

By the "Gene" resource it is possible to access to specific sections dedicated to classes of gene with relevant interest. Gene chromosome location, sequence and structural information are extracted from the NCBI Taxonomy database, used also as reference for information on the biological sources of the protein sequenced providing links to the main important biological database (KEGG). This section is still in progress.

4) The web interface

The web interface has been developed by using Microsoft ASP.NET technology, by leveraging on Framework .NET 2.0. Care has been taken to allow a simple and rapid update of the database with the inclusion of new entries.

The database is open source in order to maintain consistency with the new findings and can also be used as a guideline in order to create other sections for other relevant metabolites. At the moment, it is accessible at the following URL:

www.ylichron.it/PHA_pseudomonas_DB.

II. CONCLUSION

The tool we describe here has been developed to support lab scientists and bioinformatics to gain information and data about *Pseudomonas* species, targeting sequences of the most important classes of

compound and biotechnological interest. The way it provides for the retrieval and extraction of sequences allows the user to overcome the obstacles encountered in the integrative use of different bioinformatic resources. At the meantime, the completeness of the sequence collection allows intra- and interspecies comparison at different biological levels (genes, transcripts and proteins.

ACKNOWLEDGMENTS

This work has been performed in the frame of the project "CRESCO" (Computational Research Center for Complex Systems) co-founded by ENEA and the Italian Ministry of University and Research in the frame of "Programma Operativo Nazionale 2000-2006 Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione, Misura II.2 : Società della Informazione per il Sistema Scientifico Meridionale, Azione a : Sistemi di calcolo e simulazione ad alte prestazioni".

REFERENCES

[1] Choi C, Münch R, Leupold S, Klein J, Siegel I, Thielen B, Benkert B, Kucklick M, Schobert M, Barthelmes J, Ebeling C, Haddad I, Scheer M, Grote A, Hiller K, Bunk B, Schreiber K, Retter I, Schomburg D and Jahn D. (2007) SYSTOMONAS — an integrated database for systems biology analysis of *Pseudomonas*. Nucleic Acids Res 35:533-537

[2] Madison L, Huisman GW. (1999) Metabolic engineering of poly(3-hydroxyalkanoates): from DNA to Plastic. Microbiol. Molec Biol Reviews 63 (1): 21–53.

[3] Rehm BHA, Steinbuchel A. (1999) Biochemical and genetic analysis of PHA synthases and other proteins required for PHA synthesis. Int J Biol Macromol 25: 3–19.

[4] de Eugenio LI, Garcia P, Luengo JM, Sanz JM, Roman JS, Garcia JL, Prieto MA. (2007) Biochemical evidence that *phaZ* gene encodes a specific intracellular medium chain length polyhydroxyalkanoate depolymerase in *Pseudomonas putida* KT2442: characterization of a paradigmatic enzyme. J Biol Chem. 16, 4951–4962.

[5] Hoffmann N, Rehm BHA. (2004) Regulation of polyhydroxyalkanoate biosynthesis in *Pseudomonas putida* and *Pseudomonas aeruginosa*. FEMS Microbiol Lett 237: 1–7.

[6] Hoffmann N, Rehm BHA. (2005) Nitrogen-dependent regulation of mediumchain length polyhydroxyalkanoate biosynthesis genes in pseudomonads. Biotechnol Lett 27: 279–282.

[7] Dias JML, Oehmen A, Serafim LS, Lemos PC, Reis MAM, and Oliveira Rui (2008). Metabolic modelling of polyhydroxyalkanoate copolymers production by mixed microbial cultures. BMC Syst Biol 2008: 2:59

[8] Nogales J, Palsson B and Thiele I. (2008) A genome-scale metabolic reconstruction of *Pseudomonas putida* KT2440: *i*JN746 as a cell factory BMC Syst Biol 2:79

[9] Puchaka J, Oberhardt MA, Godinho M, Bielecka A, Regenhardt D, Timmis KN, Papin JA, Martins dos Santos VAP. (2008) Genome-Scale Reconstruction and Analysis of the Pseudomonas putida KT2440 Metabolic Network Facilitates Applications in Biotechnology. PLoS Comput Biol 4(10)

[10] Licciardello G, Catara V, Mangtani A, Casilli R, Rosato V. (2008) PseudoBioRes: una risorsa bioinformatica per il genere *Pseudomonas.* Conferenza Nazionale Italiani E-Science 2008, Book of abstract 126.

Grazia Licciardello was born in 1978. She is a molecular biologist with a good expertise in plant pathology and biotechnology thanks to a II level Post graduate Master in "Biotechnology for sustainable protection of crops and agri-food" and a PhD in "Phytosanitary technologies" at the University of Catania. From 2004 up to now, she works as researcher at Scientific and Technological Park of Sicily. She has participated to the following project: the PON

project "Utilization of waste material to develop biodegradable polymers (PHA) for agriculture and agroindustry" and the MIUR project "CRESCO, Computational Centre for research on Complex Systems".
Her main area of research is the genetic manipulation for biotechnology purposes, the detection of microbial phytopathogens by molecular methods and the study of genes involved in genetic regulation. She is author of about 30 scientific papers published in international refereed journals and presented in national and international congress.

Vittoria Catara, Associated Professor at the University of Catania. Since 1990 she has joined the research activity of Di.S.Te.F University of Catania, Italy; she cooperated in a number of projects; she has been responsible for 3 Project of the University of Catania, in a project for young researcher of Catania University and Coordinator of a British programme funded by CRUI and British council. She is involved in phytobacteriology studies and on to molecular aspects of fungal diagnosis and characterization. She collaborated at the preparation of 120 contributes among scientific publications, published on technical and scientific journals, or presented in conference and published in proceedings on the following subjects: plant diseases; molecular techniques for the diagnosis of plant pathogens; phenotypic and genomic characterization of *P. corrugata*; evaluation of resistance to biotic and abiotic factors; characterization and application of biocontrol agents; analysis of bacterial populations; Evaluation of bacteria for polyhydroxyalkanoates production; regulation of polyhydroxyalkanoates genes, Quorum sensing in Pseudomonas spp. She described new diseases from known pathogens, described a new bacterial species, P. mediterranea Catara et al (2002). She is co-author of a patent.

Recent advances of MedIGrid PSE in an LCG/gLite environment

V. Boccia[1], L. Carracciuolo[2], L. D'amore[1], G. Laccetti[1], M. Lapegna[1] and A. Murli[1]
[1]*University of Naples Federico II, vania.boccia@unina.it, luisa.damore@dma.unina.it, giuliano.laccetti@dma.unina.it, marco.lapegna@dma.unina.it, almerico.murli@dma.unina.it*
[2]*CNR, luisa.carracciuolo@cnr.it*

***Abstract*—In the context of the S.Co.P.E. national project, here we are concerned with improvements and enhancements of a medical imaging Grid-enabled framework, named MedIGrid, oriented to the transparent use of resource-intensive applications for managing, processing and visualizing biomedical images. In particular we describe an implementation of the MedIGrid PSE in the environment of the S.Co.P.E. project, based on the integration of the LCG/gLite and SCOPE-toolkit middlewares, reporting some planning strategies aimed to obtain more efficiency in data management and job submission. First we focus our attention on how to exploit the features of the new middleware environment to improve the PSE efficiency and services reliability: in particular our choices allowed to avoid:**

- **bottlenecks due to multiple and unuseful Resource Broker (WMS) transits;**
- **bound to data sizes, imposed by WMS.**

Further, we spent also some work to modify, extend and improve the underlying numerical components to introduce, at the application level, mechanisms to sustain a given level of performance and to survive to a fault during execution flow. Finally, we also devote some comments and suggestions on possible new middleware enrichments and/or integrations.

I. Introduction

THE activity, described in this paper, is partially supported by PON S.Co.P.E. project (Italian acronym for high Performance, Cooperative and distributed System for scientific Elaboration). This is a research project with two aims: the development of applications in fundamental research fields and the implementation of an open and multidisciplinary Grid infrastructure, involving several departments of the University of Naples Federico II distributed in metropolitan scale in the city of Naples. In this context we were involved in improving and enhancing a medical imaging Grid-enabled PSE, named MedIGrid, oriented to the transparent use of resource-intensive applications for managing, processing and visualizing biomedical images (see [22], [17]). The reason of the choice of the LCG/gLite middleware as reference environment is twofold: it provides some general purpose and high level services (Data and Metadata Catalogue, Resource Broker, ...) and it is the de-facto choice for the most of the Italian (S.Co.P.E. [10], PI2S2 [9], CYBERSAR [2], CRESCO [1], SPACI [11], IGI [6], ...) and European (EGEE [3], EGI [4], ...) Grid projects.

This paper is organized as follows: in Sec. II we report some details about the Grid middleware infrastructure, in Sec. III we describe the MedIGrid implementation in this environment, focusing on some planning strategies aimed to obtain more efficiency in data management and job submission; in Sec. IV two MedIGrid PSE application kernels are reported, focusing on recent modifications done to bring into applications fault tolerance and a checkpointing system. Finally in Sec. V we report conclusions, future works and some suggestions on possi-

ble middleware enrichment and/or integration aimed to support HPC applications.

II. The Grid infrastructure

As cited before, the testbed we used to develop our work is mainly the SCoPE infrastructure.

From the hardware point of view the S.Co.P.E. infrastructure is constituted by up to two thousands computational cores linked by means of InfiniBand connections and about 100 TB of disk space. These resources are distributed among some Departments of the University of Naples Federico II and are connected by a Metropolitan Area Network implemented by a 1 Gbps fibre channel technology.

From the software point of view it is essentially based on the integration of the LCG/gLite and SCOPE-toolkit middlewares. The low level middleware LCG/gLite provides the user with high level services for scheduling and running computational jobs, accessing and moving data and obtaining information on the Grid infrastructure, all embedded into a consistent security framework [18]. To allow the delevopers to integrate their applications into the Grid environment the low level middleware has to be enriched with a set of consolidated libraries on which applications rely, to form the so called "applicative middleware". In the context of S.Co.P.E. project, the *SCOPE-toolkit* fills the existing gap between low level middleware and applications: libraries (i.e. BLAS, LAPACK e ScaLAPACK for linear algebra and PETSc and FFT for scientific computation), other scientific software (i.e. GROMACS for molecular dynamics), compilers and so on, are collected in a self contained package that automizes (by means of standard installation/validation procedures) the dissemination of nedeed software on the computing resources, making applications easily integrated into the S.Co.P.E. Grid infrastructure.

III. The new PSE design

MedIGrid is oriented to the transparent use of resource-intensive applications for managing, processing and visualizing biomedical images.

The present MedIGrid applications are denoising of 3D echocardiographic images sequence and segmentation of 2D echocardiographic images.

In Fig. 1, is shown the web page and the main work sections of the PSE: after a simple authentication phase, the user has the following possibilities:

- to select the action (denoising, segmentation, etc.) to be executed on input images, configure some parameters and start the execution ;
- to monitor the execution;
- to use some tools, based on standard library as ImageJ [7], to visualize and post process output image data.

In our previous works [17], [22], some details, about a first PSE implementation in Globus environment, are reported paying particular attention to some relevant aspects as fault tolerance and performance maintenance.

Now we describe the activities we made on MedIGrid in the new underlying Grid environment (gLite instead of Globus, see Fig. 2).

This work involved both modifications to the PSE portal and the implementation of some strategies into PSE application kernels, aimed to draw PSE fault tolerance and performance maintenance.

The main activity tasks were:

- A server side modification of the Grid Portal by means of calls to functions of the LCG/gLite Java library to interface the PSE with some new collective services:
  - LFC, for a more efficient data management,
  - WMS, for a more complete and general purpose resources discovery and selection,
  - VOMS, for the management of VO access policies to the resources, and
  - MyProxy to automatic credential renew;
- The integration of some middleware features and additional code to implement such a kind of job monitoring,

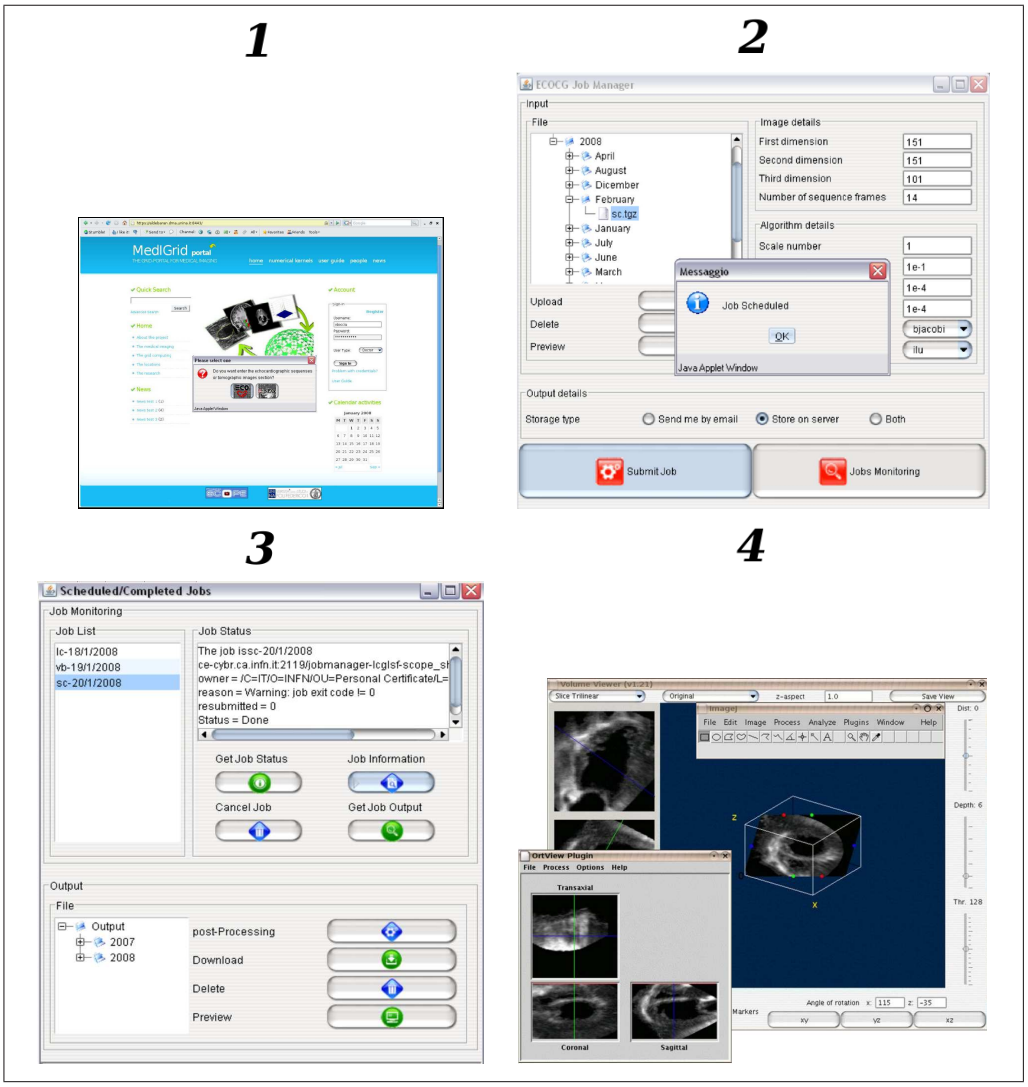


Fig. 1. MedIGrid Portal: 1) authentication and medical imaging section choice, 2) job submission, 3) job monitoring and 4) images visualization and post processing

- The introduction of an embryonal checkpointing system implemented at application level.

While we had to modify the PSE "hidden" part, we left unmodified the GUI, already complete of a satisfating number of consolidated tools for image processing parameter configuration, image visualization and post-processing (see Fig.1)

The strategies to prepare PSE to fault tolerance and performance maintenance are shortly described in the next section. In the rest of this section our attention is focused on some planning strategies aimed to obtain more efficiency in data management and job submission.

In particular our choices allowed us to avoid bottlenecks, due to multiple and unuseful Resource Broker (WMS) transits, and limitation

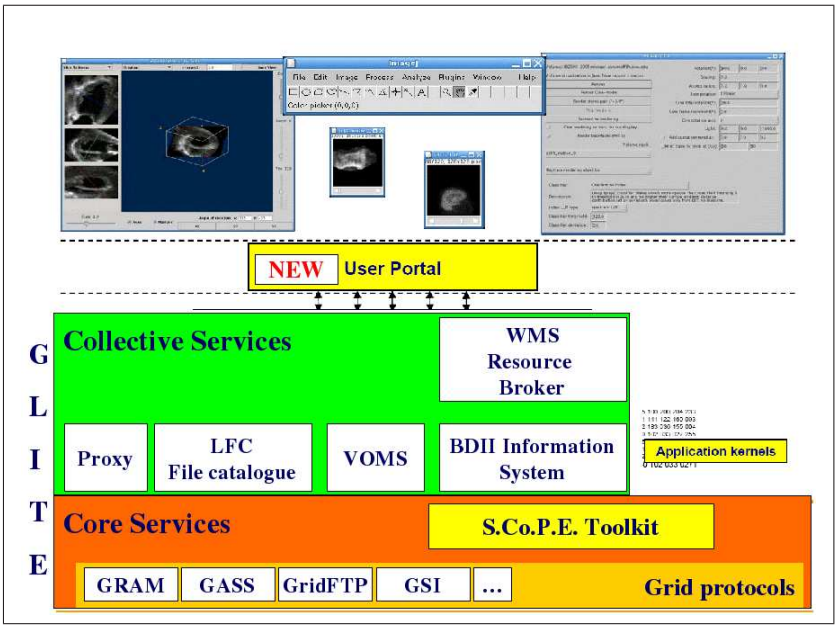


Fig. 2. New PSE architecture (gLite based)

imposed by WMS to data sizes.

To give an idea of problem size we can observe that an input sequence of only 14 ecographic small images is large more than 20*MB* and the output of the process is $20MB \times nscales$ (where $nscales$ is an input parameter related to the output quality). As the images dimension and the number of problems to be solved increase, we have to avoid unuseful paths in data transferring and redundance storage on WMS, to provide an efficient data management.

In the MedIGrid job execution schema, application binaries and data are stored on Storage Resources and registered on a Logical File Catalogue with symbolic names. Only text files containing image processing parameters are present on the UI.

At execution time, parameter files reach computational resources passing through the Resource Broker (WMS), while binaries and input data are transferred directly from Storage to Computational Resource, reducing the execution bottleneck. To reach this aim, we had to modify our numerical kernels, by means of calls to LFC/GFAL library functions: if LFC functions let us to access data transparently, by means of their logical names, GFAL tools allow efficient data transfer by the multistream GridFTP low level protocol.

At the end of the execution the output log file reaches the UI through the WMS, while the output image is directly moved into the Storage and registered onto the File Catalogue.

Summarizing, looking at the schema in Fig. 3 some main steps are performed:

- Job definition (steps 1-2): The user, by using the Grid Portal, defines the job parameters (by choosing image processing algorithm, by browsing the File Catalogue to select input data, by fixing the output data logical name, ...) through the Graphical Interface,
- Job submission (steps 3-4): The submission service on Grid Portal, submits the defined job to the Resource Broker that choices a Computing Element,
- Job execution (steps 5-6): The first step of the job script asks to LFC for application

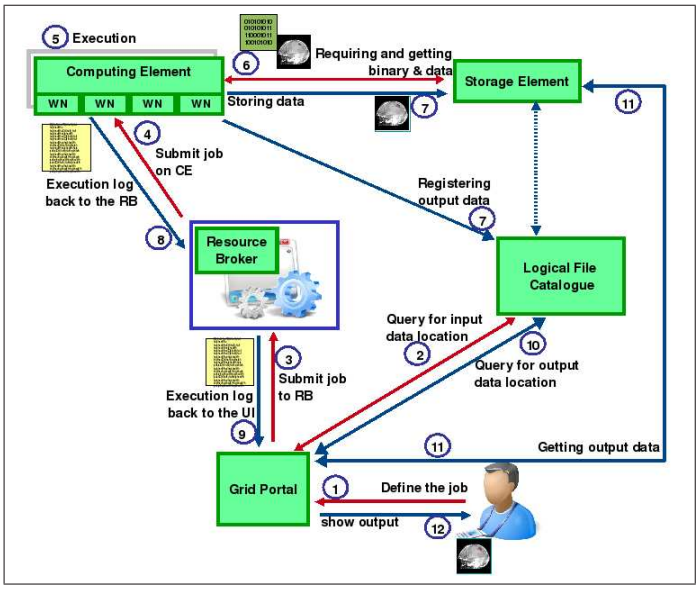


Fig. 3. Job Sumbission: MedIGrid schema

binary and input data and then gets them directly from the Storage Element,

- Data archiviation (step 7): At the end of the process the job script copies the output data directly into the SE and, at the same time, register files onto the File Catalogue with the symbolic name chosen at step 1 from the user,
- Output retrieval and visualization (steps 8-12): The execution log file moves from the Working Node to the User Interface through the Resource Broker to be visualized by the user; the user can require output files that, by means of their symbolic names, are moved from the SE directly on the Grid Portal and then visualized.

As reported above, at step 1 (see Fig. 3), the User defines, through the GUI, some requirements and parameters; on server side these ones will be used by the *job service* to construct the job in a Job Description Language format (JDL file). The job service is implemented by using objects and methods from native gLite standard library and from extra WMS-Proxy library [8].

In Fig. 4 we show an example of JDL

```
Type="Job";
 JobType="MPICH";
 NodeNumber=6;

 Executable="ECOCG.sh";
Arguments="6 scope scopelfc01.dsf.unina.it
             scoperb01.dsf.unina.it:2170
             scopese01.dsf.unina.it
             2008/February/
             sc 1e-1 1e-4 1 1e-4 150 150
             100 14";
StdOutput="ECOCG.err_out";
StdError="ECOCG.err_out";
 InputSandbox="ECOCG.sh";
 OutputSandbox="ECOCG.err_out","Denoise3d.out";
```

Fig. 4. The JDL File generated by using WMS-PROXY Java API

file generated for the denoising algorithm. We highlight (see text in bold face) some relevant attributes:

- `JobType` and `NodeNumber` used to identify a parallel job (based on MPI),
- `InputSandbox` that contains only the script file with parameters setting,
- `OutputSandbox` that contains only the job execution log files.

In Fig. 5, we report the script file code. We

```
#!/bin/sh
 # ECOCG.sh
CPU_NEEDED=$1
VO=$2
LFC_HOST=$3
LCG_GFAL_INFOSYS=$4
VO_DEFAULT_SE=$5
NAME_PATH=$6
NAME=$7
FILEN=$NAME_PATH/$NAME
# algorithm parameter setting:
# N1, N2, N3, SIGMA, SCALESTEP,
# NSCALES, RCONVERGENCE, NOF.
OUTFILE=$NAME_PATH/$16
 PROGARGS=-filename $NAME
          -scalestep $SCALESTEP
          -sigma $SIGMA -nscales $NSCALES
          -iz 1 -rconvergence $RCONVERGENCE
          -n1 $N1 -n2 $N2 -n3 $N3
          -ksp_type gmres -pc_type bjacobi
          -sub_pc_type ilu -ksp_max_it 300
          -nof $NOF
echo "Downloading binaries and input data"
echo "  ================================="
 lcg-cp --vo $VO
    lfn:/Grid/$VO/MEDIGRID/SOFTWARE/Denoise3d
    file:'pwd'/Denoise3d
 lcg-cp --vo $VO
    lfn:/grid/$VO/MEDIGRID/ECOCG/Input/$FILEN.tgz
    file:'pwd'/$NAME.tgz
tar xzvf $NAME.tgz
echo "  ================================="

echo "Executing mpirun"
echo "  ================================="
 mpirun -np $CPU_NEEDED -machinefile
           $HOST_NODEFILE 'pwd'/Denoise3d
           $PROGARGS > Denoise3d.out 2> &1
echo "  ================================="

echo "Uploading Output files"
echo "  ================================="
tar czvf $NAME.OUT.tgz Stato.txt $NAME*_*
 lcg-cr --vo $VO -d $VO_DEFAULT_SE
    -l lfn:/grid/$VO/MEDIGRID/ECOCG/Output
    /$OUTFILE.tgz file:'pwd'/$NAME.OUT.tgz
echo "  ================================="
```

Fig. 5. An example of execution script

highlight (see text in boldface) some relevant lines:

- `PROGARGS` variable that contains all the user defined parameters,
- the download of input data and binaries, from the storage to the computing resource, using `lcg-cp` command,
- the execution of the parallel application, using `mpirun` command,
- the output data transferring onto the storage and its registration into file catalogue, using `lcg-cr` command.

The Job monitoring system is implemented by combining two mechanisms:

- the *perusal* feature provided by WMS broker, and accessed by means of Java API, to retrive the job execution log files at runtime,
- a custom service that retrives partial output saved by the application and performance statistics during its execution (see checkpointing mechanism described later on).

IV. PSE APPLICATION KERNELS

As cited in the previous section, all PSE application kernels are based on the *Portable, Extensible Toolkit for Scientific Computation* (**PETSc**) library [12], [13], [14], a suite of data structures and routines that provide building blocks for the implementation of large-scale codes concerning the scalable parallel solution of scientific applications modeled by partial differential equations.

PSE applications are deeply described in [19], [20], here we focus on how these have been modified to introduce, at *application level*, some kind of checkpointing methods, combined with fault-tolerance mechanisms.

With an *application level* approach, the mechanisms enabling the fault-tolerance features in the Grid applications, are implemented in the algorithms of application kernels, so they are not demanded to the middleware, delegating the application developer to deal with the matter of appropriately choosing and saving data. On the other hand, with this strategy, the application is able to gain a higher level of efficiency in storing and retrieving the data from which performing restarting tasks.

Both the denoising and segmentation applications are based on iterative schemes and, in order to recover from a fault, they can restart from those "points" corresponding to the last "correct" computed data. So we can use a kind of *"disk based checkpointing"* method, registering those data to the LFC service, at each iteration, so that we could be able to restart with the

last computed data; at fault occurrence, application is automatically re-submitted on alternative resource accessing LFC service to obtain last computed data from which perform restarting task. The disk-based checkpointing strategy in an iterative scheme collects the following key steps:

- the last "correct" computed data is retrieved from remote storage resource by LFC service,
- the current data is computed,
- the computed data is stored on remote storage resource and registered on the LFC.

We can observe that, such kind of *disk-based* checkpointing, give us the chance to replicate data on several remote storages (as allowed by LFC/GFAL system): data are always available, even if storage and computing resources can become unavailable. This implement a storage service reliability, also paying for overhead in data transfer.

A more efficient and secure approach would require, at fault occurrence, a combined disk and diskless checkpointings: by means of sporadic and distributed disk based checkpointing we can preserve from high severity faults (as the fault of an entire cluster) while the diskless checkpointing ensures the application survival at some node fault occurrence paying a little overhead.

Thus, to use WMS authomatic re-scheduling, the application has to execute the following tasks:

1) checking the presence of a fault after each communication phase, and,
2) if a fault or a severe performance degradation is found, terminating execution with an exit non-zero code.

With the aim to develop a system for performance maintenaince, at application level, we prepared applications to return runtime performance statistics to an external process.

V. Conclusions and Future Works

This paper describes some work done to implement the MedIGrid PSE in an LCG/gLite environment. Things are much evolved, starting from our preliminary results, and, although this experience has been realized in economy of thought, features and tools now available in the LCG/gLite middleware open new scenarios for the medical community to collaborate and share resources, information and knowledge.

As mentioned in previous section, to modify, extend and/or improve the underlying numerical components, we have to introduce some checkpointing techniques, to be combined with fault-tolerance mechanisms. To introduce fault-tolerance mechanisms, we have to modify appropriately both the application and the "high level" middleware components (i.e. PETSc).

Actually our work involves the implementation of mechanisms that handle all communication faults, at level of PETSc library, by using, as message passing system implementation, FT-MPI [5] (at present the only MPI-2 standard implementation that supports communicator processor restoration).

Thus, as described in this document, we used the high level middleware mechanisms to manage in efficient way data transfer, but no much work, until now, has been devoted on data security and simple metadata management (very relevant problem if we think to medical field). In the future, we will take part to the development of middleware tools and services aimed to handle this matter in a standard way (i.e. by introducing DICOM support both at application and at middleware level).

References

[1] *CRESCO Project* Home Page. http://www.cresco.enea.it/
[2] *CYBERSAR Project* Home Page. http://www.cybersar.com.
[3] *EGEE Project* Home Page. http://www.eu-egee.org/.
[4] *EGI Project* Home Page. http://web.eu-egi.eu/.
[5] *FT-MPI:Fault-Tolerant Message Passing Interface User Manual* http://www.cs.utk.edu/ angskun/ftmpi_manual/ftmpi.ps
[6] *IGI Project* Home Page. http://grid.infn.it/igi/partners.html.
[7] *ImageJ: Image Processing and Analisys in Java.* http://rsbweb.nih.gov/ij/index.html.
[8] *Java Library for WMProxy* https://grid.ct.infn.it/twiki/bin/view/GILDA/ApiJavaWMProxy.

[9] *PI2S2* Project Home Page. http://www.pi2s2.it/.
[10] *S.Co.P.E. Project* Home Page. http://www.scope.unina.it.
[11] *SPACI Project* Home Page. http://www.spaci.it/.
[12] S. Balay, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. *PETSc users manual* Technical Report ANL- 95/11 - Revision 2.1.5, Argonne National Laboratory, 2004.
[13] S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. *PETSc Web page*, 2001. http://www.mcs.anl.gov/petsc.
[14] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith. *Efficient management of parallelism in object oriented numerical software libraries.* In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, Modern Software Tools in Scientific Computing, pages 163-202. Birkhäuser Press, 1997.
[15] G. B. Barone, D. Bottalico, V. Boccia, L. Carracciuolo, S. Pardi, M. Scognamiglio, and F. Serio. *Middleware Applicativo: lo S.Co.P.E. Toolkit.* In Universit'a degli Studi di Napoli "Federico II", editor, Proceedings of Italian National Conference of e-Science 2008, Napoli, 2008. CRUI.
[16] V. Boccia, L. Carracciuolo, L. D'Amore, G. Laccetti, M. Lapegna, A.Murli *The MedIGrid PSE in an LCG/gLite environment* International Symposium on Parallel and Distributed Processing with Applications International 2008 (ISPA '08), Workshop on High Performance and Grid Computing in Medicine and Bioinformatics (HiPGCoMB) Sydney, Australia 10-12 Dicembre 2008, ISBN: 978-0-7695-3471-8, pp.829-834.
[17] V. Boccia, L. D'Amore, M. Guarracino, and G. Laccetti. *A Grid Enabled PSE forMedical Imaging Applications: Experiences on MedIGrid.* In Proceedings of The 18th IEEE Symposium on Computer-Based Medical Systems, pages 529-536, Boston, 2005. Springer-Verlag.
[18] S. Burke, S. Campana, P. M. Lorenzo, C. Nater, R. Santinelli, and A. Sciaba. *GLITE 3.1 USER GUIDE* 2008.
[19] L. Carracciuolo, L. D'Amore, and A. Murli. *Towards a parallel component for imaging in PETSc programming environment: A case study in 3-D echocardiography.* Parallel Computing, 32:67-83, 2006.
[20] L. D'Amore, D. Casaburi, L.Marcellino, and A.Murli. *Segmentazione di sequenze di immagini digitali mediante previsione del moto.* TR 62, Dip. di Matematica e Applicazioni, Università degli Studi di Napoli Federico II, 2006.
[21] E. Laure and alt. *Programming the Grid with gLite.* Computational Methods in Science and Technology, 12:33-45, 2006.
[22] A. Murli, V. Boccia, L. Carracciuolo, L. D'Amore, G. Laccetti, and M. Lapegna. *Monitoring and Migration of a PETSc-based Parallel Application for Medical Imaging in a Grid computing PSE.* In P. Gaffney and J. Pool, editors, Grid-Based Problem Solving Environments, pages 421-432, Boston, 2007. Springer-Verlag.

PI2S2-ImmunoGrid Services Integration: a Web-Based Working Example

E. Mastriani[1] , M. Halling Brown[2], E. Giorgio[3], F. Pappalardo[4], A. Shepherd[2] and S. Motta[5]
[1]*Consorzio COMETA, Catania, Italy, mastriani@dmi.unict.it*
[2] *School of Crystallography, Birkbeck College, Malet Street London WC1E 7HX, UK, {a.shepherd, m.halling}@mail.cryst.bbk.ac.uk*
[3]*Consorzio COMETA, Catania, Italy, emidio.giorgio@ct.infn.it*
[4]*University of Catania, Catania, Italy, francesco@dmi.unict.it*
[5] *Faculty of Pharmacy and Departement of Mathematics and Computer Science, Catania, Italy motta@dmi.unict.it*

***Abstract*— We present an overview of the system architecture to implement a solution for integration of ImmunoGrid Web Services (IWS) and PI2S2 infrastructure. The main goal is to avoid a generic user to pass through all related and annoying GRID job submission steps to run the ImmunoGrid Simulator. ImmunoGrid is a simulation framework to model of the Human Immune System using Grid technologies. It integrates processes at molecular, cellular and organ levels. The access to Grid services is made by a Virtual User Interface (VUI) based on SLCERN306-VIRTUAL-UI. It ran by VM-Ware application installed on the Web server. The authentication step from IWS to the VUI is made by SSH without password or passphrase using public key authentication with an agent. The previous attempts leaded to a proof of concept of the integration. Actually the system is ready and up and now it is able to run also the graphical engine in order to visualize ImmunoGrid simulator results. As an example, last works included an upgrade step to provide tools for graphical needs. Moreover a general effort was made in order to use only open source software.**

***Index Te1ms*— COMETA, computational immunology, grid computing, interoperability, virtual user interface.**

I. INTRODUCTION

ImmunoGrid (IG) is a simulation framework to model the Human Immune System using Grid technologies. It also integrates other developed modules such as SimTriplex [1], C-ImmSim [2] and other prediction models which simulate antigen processing and presentation. The need for Grid resources has arisen from the increasing numbers of simulations to be run and the growing size of the simulation space modelled. The ImmunoGrid Web Server (IWS) was implemented in order to provide a centralised location for the development of the ImmunoGrid framework and to provide a public resource to access these resources. The IWS is a mixture of web pages containing information on the project and mechanisms to launch monitor and view IG simulations. The ability to centralise the access to the ImmunoGrid simulators avoids the need for local versions of software.

The main goal described in this paper is to equip the IWS with the capabilities to launch jobs also on the PI2S2-Grid, simplifying the submission process for the user. One of the main conventions followed during the development of IG was to allow access to a wide variety of resource through the same mechanism or interface. The IG project has developed a solution which is made up of a single web-based interface which is coupled

with a job broker. This job broker/launcher accesses PI2S2, UNICORE or AHE [3] clients via command line tools and web services via standard SOAP protocols. The main benefit of this setup is the ability to launch jobs on different Grid platforms, local resources and web services all through a single interface.

The access to PI2S2 Grid services is made by a Virtual User Interface (VUI). The virtual machine is run by VMWare application installed on the Web server. The authentication step from IWS to the VUI is made by SSH without password using public key authentication with an agent.

The credentials to use the Grid services are obtained by pk12 Gilda Certs. The integration between IWS and PI2S2-Grid is made by a pool of scripts that are able to initialise the voms-proxy for the user, create the needed environment to submit job, submit the job, wait for the "Done" status of the job and retrieve the output of submitted job.

The remainder of this paper is organised as follows. We begin with a general overview of ImmunoGrid European Project and related simulators in Section II. Then, in Section III, we explain the "Why the Grid is required" and our Grid philosophy from different points of view: the "implementation", "middlewares" and "portal". Section IV describes our web interfaces with particular focus to the Job Launcher. Section V describes the Glite-ImmunoGrid integration from the following points of view: usability, security and job monitoring.

II. IMMUNOGRID PROJECT

A. *Funding, aims, technologies*

The IG projects main task is to implement a computer model of the virtual human immune system which utilises Grid technologies. Computational models are required because immune system is complex and has combinatorial nature, experimental approaches are expensive, and there are restrictions on the experimentation that can be performed in humans. The developed set of tools is validated with experimental data and used in clinical applications for development of immunotherapies in cancer and chronic infections.

One of the unique aspects of IG is that it connects molecular level interactions (which regulate immune responses) with system level models (which study behaviour of the immune system as a whole). The driving force behind the ImmunoGrid project is to provide complementary applications for clinicians and developers of vaccines and immunotherapies. Further applications of the IG project include an enhanced understanding of the mechanisms of the immune system, providing educational interactive models of the immune system and apply grid technologies and infrastructure to complex scientific applications.

The IG project is funded by the European Commission under the contract FP6-2004-IST-4, NO 028069.

B. *Simulators*

When a foreign substance is introduced into our bodies, our immune system acts to neutralize that substance. This is a complex process involving the collective and coordinated response of approximately 1012 cells. In the effort to fit detailed experimental findings into a comprehensive model of the immune system, computer models and simulations are just beginning to play a role.

IMMSIM is an in machina cellular automata model of the humoral and cellular immune responses. This model attempts to incorporate a realistic description of the humoral side of the immune system and to examine questions concerned with clonal selection. The usual rules for cellular automata are the following:

- They consist of a discrete lattice of sites.
- They evolve in discrete time steps.
- Each site takes on a finite set of possible values.

The value of each site evolves according to the same deterministic rules. The rules for the evolution of a site depend only on a local neighbourhood of sites around it.

To face the problem of extending the IMMSIM automaton capabilities, a parallel version of the IMMSIM automaton named CImmSim, coded by means of the C language [8] and PVM as message passing library has been developed. CImmSim was designed according to criteria of openness and modularity to allow smooth upgrades and addition of new features (cells, molecules, interactions and so on) for future investigations. As a matter of fact, the aforementioned modularity has been recently exploited when the description of new types of cells, involved in cellular response, has been introduced with a reasonable effort. The current version of the code is able to simulate both the humoral response described in [9, 10, 8] and the cellular response described in [11]. In this respect it is the most advanced parallel version of the Celada-Seiden automaton. Actually, C-ImmSim consists of more than 10000 lines of C code. Thanks to a dynamic memory allocation, to the compressed output mechanism and to other tricks, it can work with a potential number of molecules (cell-receptors, peptides and epitopes) of the order of 224. The high degree of complexity of the model make it suitable to simulate different immunological phenomena, e.g., the hypermutation of antibodies, the germinal center reaction (GCR), immunization, Thymus selection, viral infections etc. C-ImmSim has now evolved to take into account other features not included in the original work of P.E. Seiden and F. Celada. In particular, since the first version, the model has been enriched to simulate HIV infection and cancer immunotherapy.

Cancer immunoprevention is based on the use of immunological approaches to prevent, rather than to cure cancer. Immunoprevention of spontaneous tumors has been explored in recent years in preclinical systems thanks to the availability of several new transgenic mouse models that closely mimic the natural history of human tumors [12]. The most thoroughly investigated model of cancer immunoprevention is the mammary carcinoma of HER-2/neu transgenic mouse. A complete prevention was obtained using a combination of three signals (the so called "triplex" vaccine) that included the specific antigen (viz. p185, the product of HER-2/neu) and nonspecific signals like allogeneic histocompatibility antigens and interleukin 12 [13].

The major issue still unresolved with the Triplex vaccine is whether or not the Chronic schedule is the minimal set of vaccination yielding complete, long-term protection from mammary carcinoma.

In order to give tackle this kind of problem the SimTriplex simulator was developed. SimTriplex [14] is based on a modification of the Celada-Seiden framework [9]. SimTriplex mimics at the cellular level the behaviour of immune cells of vaccinated as well as naive mice. This is a generalized cellular automaton and it is specialized in modelling mammary carcinoma - Triplex vaccine - immune system competition.

DTU prediction tools are an important part of the ImmunoGrid simulator. The most commonly used tools are NetMHC [4] and NetCTL [5]. NetMHC provides predictions about the binding of peptides to various MHC alleles using artificial neural networks (ANNs) and weight matrices. MHC:peptide binding predictions alone can be considered incomplete when the entire antigen processing pathway is taken into consideration. NetCTL attempts to provide a broader picture by including prediction of proteasomal cleavage, Transporter associated with Antigen Processing (TAP) Transporter efficiency as well as MHC:peptide binding. This tool provides a more complete view of the antigen processing pathway.

As mentioned above, one of the specific aims of IG is to provide educational modules on different aspects of the immune system. These can be accessed via the IWS and include modules on vaccine design, HIV, bacteria and cancer vaccine scheduling. The different approach used between the research area of the IWS and the educational modules can be seen in the contrived examples that are presented for the educational modules. These modules are designed to provide an insight into a specific immune process or subject. The modules make use of the IG simulators in a more limited fashion than the research areas of the IWS.

III. WHY DO WE NEED GRID?

The use of the grid is not always an essential requirement of computational projects. In the case of the IG project, the integration of grid technologies is extremely important for the following reasons:

A.We need the Grid:

- **To run large numbers of simulations.** It is an essential requirement of the project that the exploration of parameter space over a large number of individuals is preformed as the simulators are developed.
- **To run larger simulations**. An additional aim of the ImmunoGrid project is to increase the size of the simulation space in order to represent a more 'natural scale'. These much larger simulations require more powerful machines which can be provided via the Grid.
- **To provide diverse resources.** The size and complexity of simulations will vary considerable. For this reason, some resources will be more appropriate for certain types/sized simulations. By providing access to the Grid, we enable access to a more diverse set of resources than could be accessed without.

B.Our Grid philosophy

The philosophy by which the IG grid implementation was designed was to provide access to multiple diverse grid middlewares through a single interface. This approach effectively hides the complexity of the implementation from the user. This is achieved by centralising a job launcher on the IWS and providing uniformed mechanisms to upload, and define a job. Without this approach, the average user would need to deal with complexities such as VMware, applications servers and authentication. Our centralised approach encapsulates this complexity in the job launcher.

With this approach, the final user view of the grid implementation is as simple as using an average web form. The actual implementation of the individual middlewares is completely dealt with by the technician in charge of the IWS. Figure 1 provides an overview of the IG grid design and shows how each individual grid middleware is accessible via the job launcher (the job launcher is described in more detail in the next section).

This paper is mainly concerned with the creation of methods to allow the integration of the gLite middleware with the IWS (See section V). Other middleware's that have been integrated are the Application Hosting Environment (AHE), DESHL client [6] (UNICORE) and simple web service access. The AHE provides the ability to launch jobs on resources which are globus enabled as well as local (non-national grid) resources which are GRIDSam enabled. GRIDSam is a web service which is capable of forking jobs to local systems as well as building Resource Specific Language (RSL) requests to launch jobs on globus enabled machines. An AHE client is implemented on the IWS to allow

interaction with the AHE server. DESHL is a command line client which is able to launch jobs on UNICORE enabled machines. This is an ideal solution to allow access to group resources at the DEISA site – CINECA. A group portal certificate has been acquired to allow authentication of jobs launched via the IWS rather than authentication of individual users. In order to allow resources which do not have any grid middleware to be made available through the IWS, a simulator web service has been created. This service is able to run on an application server on the host resource and fork jobs to the local system. The simulator web service has been implemented in Java, Perl and PHP. Access to these services is available through simple SOAP protocols and PHP libraries.

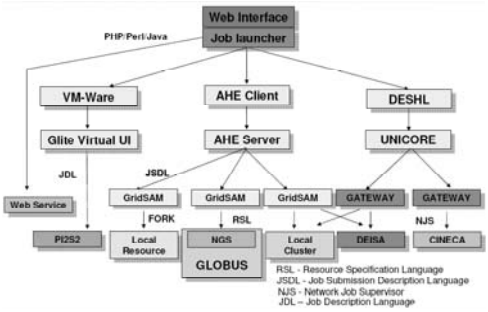


Figure 1: IG Grid design

IV. WEB INTERFACE

The web interface provides transparent access to the various different resources available through IG. The interface is presented as standard web pages and forms which are familiar to most users. The IWS is implemented in PHP, AJAX and DHTML to provide a modern look and feel to the site. The interface is used to create and prepare jobs to be launched onto the grid. The actual communication with any grid middleware is completely dealt with by the IWS job launcher. This launcher is written as a PHP script (also available as Perl scripts) which is able to interact with the various middleware clients.

This approach requires that the grid clients are completely accessible via the command line in order for the PHP job launcher to interact with them. This is the main challenge that has been faced when implemented this grid solution. Due to the diverse nature of the various clients, each interaction with the job launcher has had to be tailored to allow seamless access. Section V will describe the steps taken to enable the gLite client to be accessible over the command-line from the IWS.

When a job is defined by the user, it is then sent to the job launcher. If a resource identifier is not provided, the launcher selects an appropriate resource based on a given schedule (current default schedule: random). This scheduler can be tailored to allow specific aspects of the job definition to weight the selection of a resource. The launcher then records the details of the job in a local database and executes the appropriate command line script corresponding to the resource that is selected. The state of the job is stored in the local database along with any information required to uniquely identify that job. This allows the appropriate scripts to be called when the IWS requests the job state to be refreshed.

This centralised scheduling approach is useful to allow transparent access to multiple middleware's. It is also useful for enforcing the use of a particular resource under certain circumstances. Figure 2 shows the three different 'levels' of access available in the IWS. The resource that are available to the user, depends upon which level they are launching the job from. Jobs launched from the educational modules are only allowed access to local IWS resources as they are normally small, low-complexity jobs. Jobs launched from the developer level (requires login), have access to the full range of resources. This approach also allows the IWS to prevent unauthorised access to the grid resources by non-members.

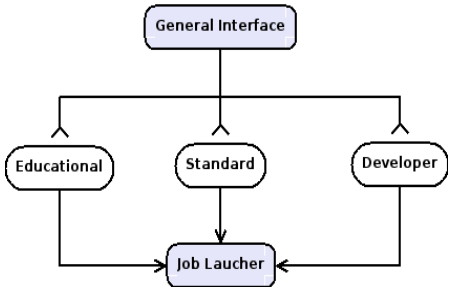


Figure 2: Different 'levels' of access available in the IWS

V. GLITE-IMMUNOGRID INTEGRATION

In this section we describe the adopted solution to integrate IWS with PI2S2-G. As shown in the Figure 3, the end user would be able to start his simulation via IWS, submit it to the PI2S2 infrastructure and retrieve movie output.

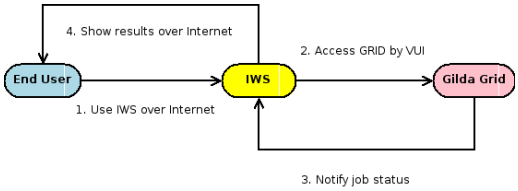


Figure 3: Use case scenario

To do this, IWS has to be able to connect to a Glite User Interface. For simplicity, we chose to customize a Virtual User Interface (VUI) that use the "glite-wms*" commands. We configured the VUI (hosted by the web server) to submit the jobs to the Gilda Grid environment. In details, the User Interface is a Virtual User Interface based on SLCERN306-VIRTUAL-UI where the *vomscert, CA_Gilda* and the *glite_wms.conf* are opportunely configured. It ran , for simplicity, by VM-Ware application. As shown in Figure 4, the system architecture is simple. The IWS, available from internet, hosts a VM-Ware server which runs the Virtual User Interface in a private class ip address. The web server machine also hosts a disk space to each user enabled to this service.

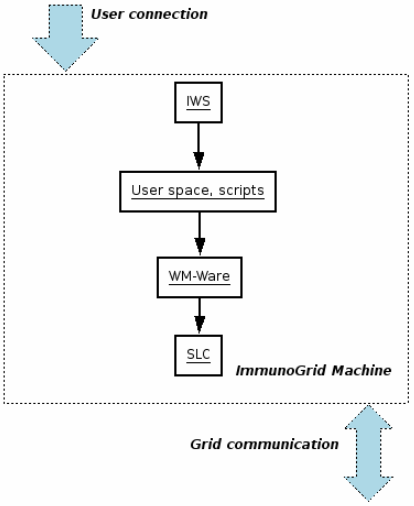


Figure 4: Integration system architecture

The authentication step between IWS and SLC is made by SSH without password or pass-phrase using public key authentication with an agent. The secure communication from the SLC to the Grid is granted by the use of Gilda PKCS12 certificates. The user which would use this service needs only two steps: obtain a Gilda certificate (with a simple online request) and obtain access (login/password) to IWS. All the configuration steps are reminded to the IWS's sysadmin that enables the user and installs the scripts to submit the job, monitor its status and retrieve the output. The integration between IWS and PI2S2-G is made by a collection of scripts that are able to: initialize the voms-proxy for the user, create the needed environment to submit the job, submit the job, wait for the "Done" status of the job showing its current status and retrieve the output of the job. Notice that the production of simulation movies needs of Octave, Gnuplot, mpeg2codec, but all is reminded to the configuration step. In this case, we used the "job submission" approach to install and configure all packages, accessing a simple user with associated an "admin" role. As you can see, the jdl file is very simple and this shows the linearity of our approach:

```
Type="Job";
JobType="Normal";
Executable = "/bin/sh";
StdError = "stderr.log";
StdOutput = "stdout.log";
Arguments = "run_simulation.sh";
InputSandbox = {"datafile", "Only_CG",
"MovieCancer_Octave.m","MovieNecrotic_O
ctave.m", "MovieNutrient_Octave.m",
MovieTumor_Octave.m,"run_simulation.sh"
, "simcgrowth",
"TimesNewRomanBoldItalic.ttf",
"TimesNewRomanBold.ttf", "TimesNewRo
manItalic.ttf", "TimesNewRoman.ttf"};
OutputSandbox = {"stderr.log",
"stdout.log", "movies_out.tgz",
"png_out.tgz", "dat_out.tgz"};
RetryCount = 0;
Requirements=Member("VO-gilda-
OCTAVE_3_2_0",other.GlueHostApplication
SoftwareRunTimeEnvironment);
```

The execution file sets the Grid-user environment and runs all "simulation" steps.

From the user point of view (after the security and authentication steps), he is able to run a SimTriplex simulation submitting a job to the Gilda-Grid. During this process, the status of the job is shown to the user as displayed below.

Starting Job Status log
Job Status is: Waiting
Job Status is: Scheduled
Job Status is: Scheduled
Job Status is: Running
Job Status is: Running
Job Status is: Done
Finished Job with status: Done

The test results and the logs of the simulation are saved in a compressed file and then they are copied (via secure mode) to the IWS user space. The log file, as showed below, contains all information useful to understand the job submission process.

From the "system architecture" point of view, the process is also simple, as the following flowchart-diagram shows.

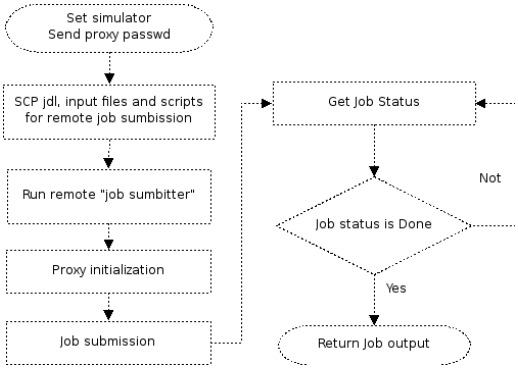


Figure 5: Flow chart

In details, once the user has configured his simulation and has sent his proxy password (over https connection to the IWS), the files useful to the test and environment configuration are copied (in secure mode) from the IWS user space to the SLC-user space. Then the simulation is started with the execution of a remote SSH command from the IWS to the SLC machine. After the end of the "job submission" procedure, it will check (with a configurable time delay) for the job status until "Done" is reached. Then the log and output files are retrieved.

VI. Conclusion

The current work shows that we have utilised and amalgamated multiple GRID solution, with particular focus to gLite integration. This approach provides both flexibility (access to widest range of resources) and redundancy ("future proofing"). The final result is that with a small configuration work from the sysadmin, the user is able to submit SimTriplex job, to monitor the status of the simulation and retrieve the output and log without any extra work. With additional development, we will also be able to submit MPI jobs to these resources.

References

[1] PL Lollini, S Motta, F Pappalardo, "Discovery of cancer vaccination protocols with a genetic algorithm

driving an agent based simulator", *BMC Bioinformatics*. 20;7:352. Jul 2006

[2] V. Baldazzi, F. Castiglione, M. Bernasch, "An enhanced agent based model of the immune system response", *Cell Immunol*., 244(2):77-9 2006

[3] P. V. Coveney, R. S. Saksena, S. J. Zasada, M. McKeown, S. Pickles, "The Application Hosting Environment: Lightweight Middleware for Grid-Based Computational Science", *Computer Physics Communications*,. 176(6): 406-418

[4] S. Buus, S. L. Lauemoller, P. Worning, C. Kesmir, T. Frimurer, S. Corbet, A. Fomsgaard, J. Hilden, A. Holm, and S. Brunak, "Sensitive quantitative predictions of peptide-MHC binding by a 'Query by Committee' artificial neural network approach". *Tissue. Antigens*., 62:378–384, 2003.

[5] Larsen M.V., Lundegaard C., Kasper Lamberth, Buus S,. Brunak S., Lund O., and Nielsen M, "An integrative approach to CTL epitope prediction. A combined algorithm integrating MHC-I binding, TAP transport efficiency, and proteasomal cleavage predictions.", *European Journal of Immunology*. 35(8): 2295-303. 2005

[6] T. M. Sloan, R. Menday, T. P. Seed, M. Illingworth, A. S. Trew, "DESHL--Standards Based Access to a Heterogeneous European Supercomputing Infrastructure" *Proceedings of the Second IEEE International Conference on e-Science and Grid Computing*, Page 91 2006

[7] P. Seiden, *personal communication*

[8] F. Castiglione, M. Bernaschi, and S. Succi. Simulating the immune response on a distributed parallel computer. *Int. J. Mod. Phys. C* **8**:527–545, 1997.

[9] F. Celada, and P. Seiden. A computer model of cellular interaction in the immune system. *Immunology Today* **13**:56–62, 1992.

[10] P. Seiden, and F. Celada. A model for simulating cognate recognition and response in the immune system. *J. Theor. Biol.* **158**:329–357, 1992.

[11] M. Bezzi, F. Celada, S. Ruffo, and P. Seiden. The transition between immune and disease states in a cellular automaton model of clonal immune response. *Physica A* **245**:145–163, 1997.

[12] Di Carlo, E., M.G. Diodoro, K. Boggio, A. Modesti, M. Modesti, P. Nanni, G. Forni, and P. Musiani, *Analysis of mammary carcinoma onset and progression in her-2/neu oncogene transgenic mice reveals a lobular origin.* Laboratory Investigation, 1999. **79**(10): p. 1261-1269.

[13] Lollini, P.-L., C. De Giovanni, T. Pannellini, F.C. Guido, and F. Nanni, *Cancer Immunoprevention.* Future Oncology, 2005. **1**(1): p. 57-66.

[14] Pappalardo, F., P.-L. Lollini, F. Castiglione, and S. Motta, *Modeling and Simulation of Cancer Immunoprevention Vaccine.* Bioinformatics, 2005. **21**(12): p. 2891-2897.

Dr. Emilio Mastriani

During my studies, I have carried out several researches in the Computer Security and Artificial Intelligence fields, obtaining relevant published results. I have worked in Operating System Security, producing "Piranha Audit". I am currently site administrator of the Grid-node of Department of Mathematics and Computer Science (University of Catania). I also offer support for the process of adapting scientific and industrial applications so that it can run over the Grid infrastructure. I am specializing on the use of the Grid technology and my latest courses are: *5th International Summer School on Grid Computing, 3rd DEISA Training Session* .I am also studying models, mathematical methods and computer simulation of cancer growth and therapy. I am a member of: *PI2S2 project, CI&IG research group and IMMUNOGRID EC project.*

Dr. Mark Halling-Brown

My research interests are focused on the areas of bioinformatics, computational immunology and Grid computing. I am involved in the following areas: the development of computational techniques for large scale analysis of immune molecules and the application of computational techniques in vaccine design, development of tools and environments that enhance and utilise distributed computing and workflow, development of generic techniques to perform context specific searches of journal papers, use of molecular dynamic techniques to further T-cell epitope prediction and development of techniques to predict promiscuous minor histocompatibility antigens.

Performance of ViralPack applications on PI2S2 Grid infrastructure

A. Lombardo[1,2], G. Lanzalone[3], A. Muoio[4], Roberto Barbera[4,5] and M. Iacono-Manno[5]
[1]*Parco Scientifico e Tecnologico della Sicilia, Z.I. Blocco Palma I, Stradale V. Lancia 57 95121, Catania, alelomba@unict.it*
[2]*Dipartimento di Scienze e tecnologie Fitosanitarie, Università di Catania, Via S. Sofia 100, 95123, Catania*
[3]*Università degli Studi "Kore"di Enna, Cittadella Universitaria, 94100, Enna*
[4]*Università di Catania, Dipartimento di Fisica ed Astronomia, Via S. Sofia 64, 95123, Catania*
[5]*Consorzio COMETA, Via S. Sofia 64, 95123, Catania*

***Abstract*— Computational Grids are emerging as a popular paradigm for heavy computing in life sciences. The aim of the present study is the development of high level computational tools for Bioinformatics, collected into a software package named ViralPack. It can be used to investigate the factors involved in virulence and dispersion of some plant viruses**
such as recombination, structural motifs and phylo-genetic relationships. The selection focused the most economically relevant ones: *Cucumber mosaic virus* (CMV), *Tomato yellow leaf curl virus* (TYLCV), *Tomato yellow leaf curl Sardinia virus* (TYLCSV), *Tomato spotted wilt virus* (TSWV) and *Citrus Tristeza virus* (CTV).
The applications have been implemented into GENIUS, a web portal acting as a friendly user-interface to the Grid based on the Enginframe framework.
The possibility of studying by means of an innovative Bioinformatic approach, the factors regulating the spread of these viruses, their structural genomic features and the events that determine their degree of virulence, become essential to withstand possible epidemic events. The possibility to build a database with the host plants so far reported in order to detect possible "host-jumps", is a further weapon. The process speed-up together with the integration of the whole phylo-genetic analysis into a coherent and easy-to-use frame, will lead to a remarkable progress in such investigations.

***Index Terms*— Grid, plant virus, GENIUS, bioinformatic.**

I. INTRODUCTION

Biotechnology such as genomics, gene sequencing and high-throughput screening are creating massive volumes and multiple sources of biological and chemical data. However, the volume of data and the processing power required to analyze it, is threatening to create a bottleneck that might hamper the growth of biotechnology itself. To date, the HPC resources required to store, manage and analyze such huge data volumes have been only available to large companies and research institutes [1]. Therefore, Grid computing offers an opportunity to address computational problems in the sciences by leveraging available and affordable resources to create a virtual scientific environment [2].

At the present major efforts are directed to new biological and biomedical technological platforms in connection with HPC and Grid technology that are particularly useful for the Bioinformatics applications or e-Health.

This paper is an outcome of an interdisciplinary working group for the development and "porting" on Grid of some useful bioinformatics applications to study the virulence and the spread of viruses in plants. The viruses selected as testbeds, *Cucumber mosaic virus* (CMV), *Tomato yellow leaf curl virus* (TYLCV), *Tomato yellow leaf curl sardinia virus* (TYLCSV), *Tomato spotted wild virus* (TSWV) and *Citrus Tristeza virus* (CTV), are, at least from an economic point of view, the causal agents of some of the most important pathologies for plants cultivated in our region.

A. The testbed viruses

Cucumber mosaic virus (CMV, family *Bromoviridae*, genus *Cucumovirus*) is one of the most important world-wide virus, infecting more that 1,000 species in over 70 families of mono- and dicotyledonos plants, especially in temperate regions. This virus, which is transmitted by aphid in a non-persistent manner by almost 60 aphid species, causes severe diseases in ornamentals, legumes, vegetables, tobacco and other important crops, such as medicinal and aromatic plants. In Italy, CMV is endemic in all horticultural and/or ornamental areas (i.e. Sicilian and Liguria regions), where it causes damages of both field-grown vegetables (tomato, pepper, cucurbits, etc.) and greenhouse crops (green and flower pot plants, shrubs, cut flowers, etc.) [3];

Citrus tristeza virus (CTV), a member of the genus *Closterovirus*, is the most economically important virus affecting citrus production worldwide. The virus is phloem limited and transmitted by aphids in a semipersistent mode and by infected buds. CTV virions are flexuous filaments containing two capsid proteins of 25 and 27 kDa that cover about 95 and 5% of virion, respectively. CTV genome is a monopartite positivesense RNA of ca. 20 kb containing 12 open reading frames (ORFs) encoding at least 19 proteins [4].

Citrus tristeza virus (CTV) has been reported in Italy since 1953, mostly on imported citrus varieties. The first outbreak on local sweet orange varieties, as Tarocco and Navelina, in a relevant citrus area of Sicily occurred many years later. Surveys of the last five years show that CTV is differently diffused in nearly all the citrus areas, with disastrous results when trees are grafted onto sour orange [5].

Tomato yellow leaf curl virus (TYLCV) (*sensu lato*) is the name given to a complex of virus species that cause a disease that is a threat to tomato (*Lycopersicon esculentum*) production in many tropical and subtropical regions worldwide and causes up to total yield loss. Symptoms of the disease consist of a more or less prominent upward curling of leaflet margins, reduction of leaflet area and yellowing of young leaves, together with stunting and flower abortion. Infection results in a general decrease of plant growth and reduced yields, and production is almost entirely lost if plants are infected during early growth [6].

Tomato spotted wilt virus (TSWV), the type member of the genus *Tospovirus* in the family Bunyaviridae, is by far the most economically important tospovirus with a broad host range and worldwide distribution. The pleomorphic particles of TSWV contain a tripartite genome of negative or ambisense polarity [7].

B. The applications

TOPALi [8] is a Java graphical analysis application that allows the user to identify recombinant sequences within a DNA, or RNA, multiple alignment. TOPALi allows a choice of three statistical methods to predict the positions of breakpoints due to past recombination. The breakpoint predictions are then used to identify putative recombinant sequences and their

relationships to other sequences. In addition to its sophisticated interface, TOPALi can import many sequence formats, estimate and display phylogenetic trees and allow interactive analysis and/or automatic HTML report generation.

ClustalW-MPI [9] is a tool for aligning multiple protein or nucleotide sequences. The alignment is achieved via three steps: pairwise alignment, guide-tree generation and progressive alignment. ClustalW-MPI is a distributed and parallel implementation of ClustalW. All three steps have been parallelized to reduce the execution time.

C. Performance of ViralPack

An important aspect of this implementation is the speed-up of the execution time to generate the output files. Fig. 1 and fig.2 compares the execution times for a TOPALi application running respectively on a remote web server and on the Grid. The method used was the Probabilistic Divergence Measure (PDM) with a 10 nucleotides step and 500 nucleotides windows. The CTV genomes consist of about 20 Kbp while the CMV segment 3 of about 2,2 Kbp.The simultaneous run of several jobs is faster on the Grid compared to the other platforms.

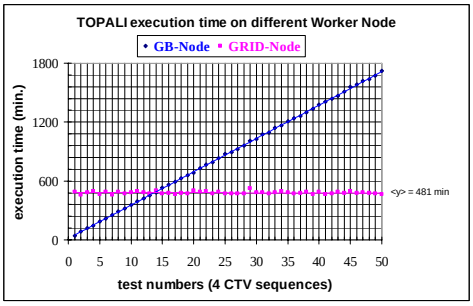


Figure 1. Simultaneous submission of a 4 CTV complete genome sequences alignment for detection of recombination by TOPALi, PDM method, on different infrastructures.

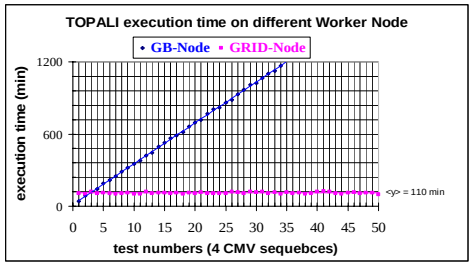
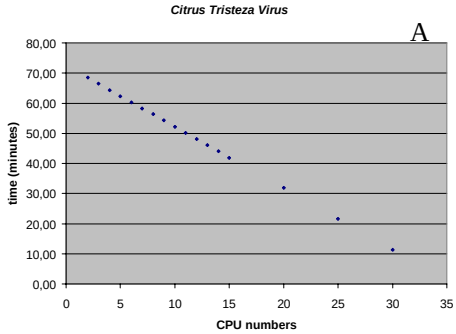
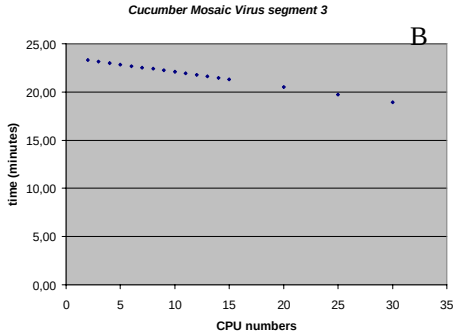
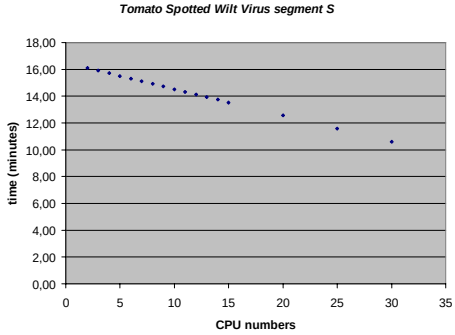


Figure 2. Simultaneous submission of a 4 CMV segment 3 sequences alignment, for detection of recombination by TOPALi, PDM method, on different infrastructures.







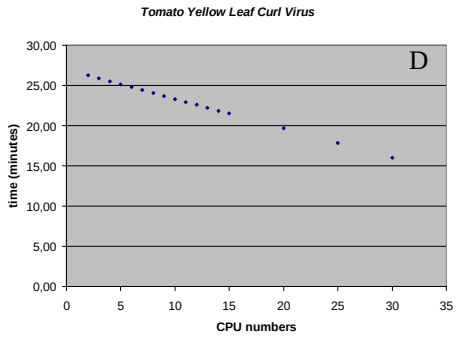


Figure 3. Mean execution time versus the number of processors for the ClustalW application. A) alignment of 10 CTV complete genomes; B) alignment of 84 CMV segments 3; C) alignment of 41 TSWV segment S; D) alignment of 56 TYLCV and TYLCSV complete genomes.

Figure 3 shows the multiple alignments of several virus sequences with different sequences length that range from about 20 Kbp of *Citrus tristeza virus* to the 2,2 Kbp of the segment S of *Cucumber Mosaic Virus* (TSWV segment s: 3 Kbp; TYLCV: 2,8 Kbp) . The speed-up process depends on the length and the number of sequences and is more evident for the longest sequences where the analysis time is reduced by a factor of about 6 with 30 CPUs.

D. User Interface

Clustalw and TOPALI applications have been implemented into Genius [10], a web portal acting as a friendly user-interface to the Grid, created in a partnership between INFN and NICE and based on the Enginframe framework using standard techniques adopting Java, XML, HTML, XSL and HTTP languages. Bash and JDL scripts have been also used in this implementation.

Enginframe is a distributed SW tool based on a Web server, and one or more Agent processes that can interact with it through XML messages. This software allows a secure login by Internet browser.

We have successfully tested the graphical interface with Internet Explorer, Firefox and Opera browsers.

The web interface of Enginframe is made by GENIUS, i.e., Grid Enabled web eNvironment for site Independent User job Submission. The goal of GENIUS is to hide the complexity of the middleware to the novices.

Figure 4. The figure shows the home page of the Genius web interface to the Grid.

II. CONCLUSION

An important aspect of this implementation is to optimize the execution time to generate the output files (for instance, genetic trees). Preliminary results are encouraging. The result shows the mean execution time versus the number of processors for the ClustalW application.

It has been demonstrated that thanks to the elevated speed of calculation of GRID it is possible to carry out the analysis of entire viral genomes, or of a very high number of sequences in "reasonable" times, reducing them, currently, of approximately a factor of ten. Hence, a new road has been started, encompassing a future, and no longer just feasible, methodology vision: an integration of visions and information, and therefore integrated know-how of agronomists, molecular biologists and IT experts.

ACKNOWLEDGMENTS

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] P. Donachy, Harmer T.J., Perrott R.H. *et al.* Grid Based Virtual Bioinformatics Laboratory. *Proceedings of the UK eScience All Hands Meeting 2003,* 111-116.

[2] Y.M. Teo,Y.K. Ng and X. Wang, *Progressive Multiple Biosequence Alignments on the ALiCE Grid,* Proceeding of the 6th International Conference on High Performance Computing for Computational Science, Springer-Verlag Lecture Notes in Computer Science Series, Spain, June 28-30, 2004.

[3] M.G. Bellardi, S. Davino, A. Lombardo, M. Davino. Biological and molecular characterization on the basis of the movement protein gene of *Cucumber mosaic virus* isolates from medicinal and aromatic plants in Italy. P*hysiological and Molecular Plant Pathology* (2008) (submitted).

[4] L. Rubio, M.A. Ayllon, P. Kong, A. Fernadez, M. Polek, J. Guerri, P. Moreno and B.W. Falk. Genetic Variation of *Citrus Tristeza Virus* Isolates from California and Spain: Evidence for Mixed Infections and Recombination *Journal of Virology* 2001, p. 8054–8062 Vol. 75, No. 17S.

[5] S. Davino, L. Rubio and M. Davino. Molecular analysis suggests that recent Citrus tristeza virus outbreaks in Italy were originated by at least two independent introductions.*European Journal of Plant Pathology* (2005) 111: 289–293.

[6] E. Moriones, J. Navas-Castillo. Tomato yellow leaf curl virus, an emerging virus complex causing epidemics worldwide. *Virus Research,* 71 (2000) 123–134.

[7] R.A. Naidu, C.J. Ingle, C.M. Deom and J.L. Sherwood. The two envelope membrane glycoproteins of Tomato spotted wilt virus show differences in lectin-binding properties and sensitivities to glycosidases. *Virology* **319** (2004) 107–117

[8] I. Milne, F. Wright, G. Rowe, D.F. Marshal, D. Husmeier e G. McGuire, 2004. TOPALi: Software for Automatic Identification of Recombinant Sequences within DNA Multiple Alignments, *Bioinformatics* 20 (11): 1806-1807.

[9] Li K.B., 2003. ClustalW-MPI: ClustalW analysis using distributed and parallel computing. *Bioinformatics Application Note*. 19 (12): 1585–1586.

[10] Visit the EnginFrame web site http://www.enginframe.com.

A. Lombardo was born in 1971. He is a molecular biologist with a P.h.D in "Phytosanitary technologies" on biotechnological applications in plant pathology at the University of Catania.
He is member of the steering committee of the italian technological platform "IT-Plants for the future", member of the scientific committee, challenge conversion, of the italian technological platform "Biofuels Italy".
Since 2005 he is responsible of the laboratory of biotechnology, genomic analysis and varietal correspondence (Science and Technology Park of Sicily) and since 2008 responsible of the laboratory of phytopathology (Science and Technology Park of Sicily). He was involved in several Interregional, national and regional projects. He is author of about 25 scientific papers published in international refereed journals and presented in national and international congress and 1 patent.
His main research areas are the development of expression vectors for plants, yeasts and bacteria, bioinformatic applied on plant viruses, variety correspondence, molecular characterization of plants, yeasts and viruses.

G. Lanzalone is nuclear physicist with a P.h.D in "Physics" on Heavy Ion Multifragmentation at Fermi's Energy at University of Study of Catania. His actual position is Research Associate in University of Study of Enna (Italy). His primary interest is in the properties of nuclear matter under extreme conditions of excitation, density, shape and N/Z. To probe the behavior of matter under these conditions a wide variety of methods have been used from mass asymmetric, high energy light heavy-ion reactions to Fermi energy collisions of more mass symmetric heavy-ions to ternary fission.

He is author of about 50 scientific papers published in international refereed journals and presented in national and international congresses.
He has recently extended his research interests towards the domain of plasma physics and to Grid applications.

Dr. Carmelo Marcello Iacono Manno, PhD, has been working for several years with the Data Acquisition Group of the Southern National Laboratory for Nuclear Physics (Catania, Italy) developing Data Acquisition and Control Systems for various experiments such as CHIMERA.
He developed software tools for data acquisition and analysis, complex software applications for instrument control and neural networks simulations for best fitting the non–linear response curves of CsI scintillation detectors.
Since 2005, he joined the activities of the Grid group working at the Catania Section of the National Institute for Nuclear Physics, focusing on massive parallel computing and supporting the integration of many heterogeneous applications on the COMETA e–Infrastructure. He also organized various training and dissemination events and contributed many proposals applying to the EU Framework programs. He published several papers about data acquisition systems, neural networks and Grid computing on conference proceedings and international journals.

Brain Functional Synchronization on Grid

F. Di Grazia[1], F. Sapuppo[1], C. Pistagna[2], D. Shannahoff-Khalsa[3], L. Fortuna[1], M.Bucolo[1]

[1]*Dipartimento di Ingegneria Elettrica, Elettronica e dei Sistemi, Università degli studi di Catania, Italy*
federica.digrazia@diees.unict.it
[2]*Dipartimento di Matematica e Informatica, Università degli studi di Catania, Italy*
[3]*Institute for Nonlinear Science, University of California, San Diego, La Jolla, California, USA*

***Abstract* — The functional connectivity of the brain has been studied here using the knowledge from two different scientific fields. The methods of Synchronization Likelihood (SL) and network theory are applied to magnetoencephalography (MEG) data in an effort to analyze the brain as a complex network. SL is time consuming and complex. Therefore GRID architecture has been used to speed up the calculation of SL matrices for very large datasets. These studies show an interesting small-world phenomenon in functional connectivity. Network and head-map images of the results are presented.**

***Index Terms*— Biopotentials, GRID architecture, network theory, small world network.**

I. INTRODUCTION

THE complexity of brain dynamics has been widely investigated and numerous techniques for the measurement and characterization of cerebral physiological parameters have been developed with the aim of understanding the brain through the use of signals or images [1-3]. The synchronization of brain dynamics in different areas in relationship to different stimuli or the subject's activity has a proved useful analytical approach when comparing normal and pathological subjects [21-24]. When working on very large experimental spatio-temporal datasets that require complex analytical methods, the use of a conventional pc became unreasonably burdensome. In our experimental study the signals obtained using a 148-channel whole-head instrument, were recorded from a subject performing a yogic breathing exercise of 51 minutes [26].

Six data sets (3 pairs) were obtained from three two-day experiments that were repeated with a time lag of approximately one month. The use of GRID technologies has been employed to help solve the computational burden of analyzing 148 channels where one channel alone consists of 15,240 samples/minute (sampling rate of 254Hz), and 51 min of data per experiment where we combine the theory of networks with the SL method [4,9]. Using a core duo 1,66 GHz 1G of Ram it was necessary to reduce the spatial dimension to 25 channels with a sub sampling step (one every 16 samples), and this took 20 days to achieve the result for one data set. The jobs run using GRID computing have furnished, in a shorter time, and have yielded matrices of pair wise SL values that represent the connectivity between couplets of MEG channels, that are then converted to un-weighted graphs and can be used for representing networks. These networks were then characterized by parameters like the cluster coefficient, path length L and degree, with an effort to investigate small world like structures. These parameters reflect an optimal situation associated with rapid synchronization and information transfer, minimal wiring costs, the balance between local processing and

global integration, all of which are important for synchronizing the neural activity between different brain regions [5,7,8,12].

Anatomical studies suggest that neural networks, ranging from the central nervous system of C. elegans [6] to cortical networks in the cat and macaque, may be organized as small-world networks [28]. Lago-Fernandez et al. showed [11] that neural network models with small-world structures facilitate a fast system response and the emergence of coherent oscillations. Sporns and Tononi used a genetic algorithm [13] to select artificial neural networks with maximal entropy, integration or complexity.

The aim of the work here is to investigate different topologies as smallworld architectures that reflect situations associated with synchronization and information transfer between different brain regions. Therefore the starting point of this study is not the neural network, but with the MEG data. The architectures emerge from the data analyses.

II. CASE STUDY

Recordings were made using a whole-head 148-channel MEG instrument (4-D Neuroimaging, San Diego, California) located at The Scripps Research Institute (La Jolla, CA). Each of the 148 pick-up coils in this instrument is a 2-cm diameter magnetometer, with a distance of 2.2 cm between coils, center to center. Each coil is connected to a SQUID that produces a voltage proportional to the magnetic field radial to the head, resulting in preferential sensitivity to neural electrical sources tangential to the surface of the scalp emanating from cortical sulci. This MEG system is contained in a magnetically shielded room that helps reduce the contribution of magnetic fields from more distant sources, and this significantly increases the signal-to-noise ratio and improves the ability to detect deeper signal sources in the brain. Trained MEG technicians positioned the subject, applied electro-oculogram leads, and performed head shape digitization. An individual was employed who is highly trained with yogic breathing techniques and in being a subject for MEG studies. Head-shape was digitized, based on known locations on the subject's head (tragus of left and right ears and nasion). Head shape data is for later co-registration between measurement coil locations, electrode locations, and scalp landmarks. Eye movements were recorded with electrodes placed above and below the right eye. Electrode impedances were set below 5 kohms. MEG data was recorded with a sampling rate of 254 Hz, with an analog filter band pass of 1 to 100 Hz.

Yogic protocol. The subject was recorded while reclining and supported at 45 degrees. He followed a well-practiced protocol that involves a 10 minute resting baseline recording (rest phase I), followed by a 31 minute exercise recording phase, followed by a second 10 minute resting recording (rest phase II), see Figure 1. The three phases are separated by a one-minute recording pause. The exercise phase consists of selectively breathing through only one nostril (using a plug for the other side, with both arms resting in the lap) at a respiratory rate of one breath per minute for 31 min that uses a repeating pattern of 15s slow inspiration, 15s breath retention, 15s slow expiration, and 15s breath hold out) [14,25,26].

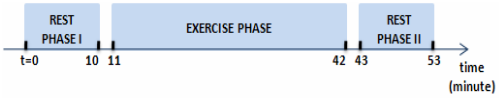


Figure 1. *Yogic Protocol Timeline.*

On day one the technique employed the left nostril as the active nostril, and on the following day, the same pattern was employed using the right nostril. This approach is used to study the potential differential brain effects that may result from these two unique meditation

techniques, and to help insure that the effects of one technique do not carry over into the effects of the other that may occur if both techniques were practiced on the same day. The use of this breath pattern using the left nostril has shown to be an effective treatment for obsessive compulsive disorder [26].
Three two-day experiments are repeated with a time lag of approximately one month. The date and left or right nostril recordings are presented in Table 1.

Table 1. *Experimental data sets over a two-month time period*

Left nostril	Right nostril
June 13, 2000 -5	June 14, 2000 - 6
July 18, 2000 -7	July 19, 2000 - 8
August 8, 2000 -9	August 9, 2000 - 10

III. METHODS

The developments in the theory of complex networks have motivated new applications in the field of the neurosciences [15]. Recent studies have led to important results in understanding the relationship between the structural properties of networks and the nature of dynamics taking place with these networks [16]. The study of models of neural networks, anatomical connectivity, and functional connectivity, based on fMRI, EEG and MEG, have been determined by using graph spectral analysis [12]. This suggests that the human brain can be modeled as a complex network, and may have different structures both at the level of anatomical as well as functional connectivity.

In this paper the SL method has been used to characterize the functional interactions as ''functional connectivity,'' by performing the measure of statistical interdependencies between signals of brain activity. The underlying assumption is that such correlations to some degree reflect the functional interactions between different brain areas.

A. Synchronization Likelihood methods

Let us consider M simultaneously recorded time series x_k with length of N, where k denotes channel number (k = 1, . . .,M). The first step in the computation of the SL is to convert the series related to each channel k in a matrix X_k in which the rows are the time series of state space vectors obtained using the method of time delay embedding [17], where L is the time lag, m the embedding dimension and $x_{k,i}$ represents the starting point of the series where i = 1,2,..., N-(m-1)*L as reported in equation (1).

$$X_{k,i} = (x_{k,i}, x_{k,i+L}, x_{k,i+2L}, \ldots, x_{k,i+(m-1)xL}) \quad (1)$$

Assuming two generic different channels and their related matrices X and Y, SL is defined as the conditional likelihood that the distance between Y_i and Y_j is smaller than a cut-off distance r_y, given that the distance between X_i and X_j is smaller than a cut-off distance r_x. In the case of maximal synchronization, this likelihood is equal to 1; in the case of independent systems, it is a small number P_{ref}, different from zero. This number is the likelihood that two vectors, randomly chosen, Y_i (or X_i) are closer than the cut-off distance r. In practice, the cut-off distance is chosen such that the likelihood of random vectors being close is fixed at P_{ref}, which is assumed to be invariable for Y_i and X_i. To understand how P_{ref} is used to fix r_x and r_y, it is necessary to consider, at first, the correlation integral of a specific channel k, identified by X:

$$C_r = \frac{2}{N(N-\omega)} \sum_{i=1}^{N} \sum_{i+w}^{N-\omega} \theta(r - |X_i - X_j|) \quad (2)$$

Here the correlation integral Cr is the likelihood that two randomly chosen vectors are closer than r. The |.| operator represents the Euclidean distance between the vectors, N is the number of vectors, w is the Theiler

correction for autocorrelation [19], and θ is the Heaviside function: $\theta(X)=0$ if $X>0$ and $\theta(X)=1$ if $X<0$. At this point r_x is chosen such that $C_{rx} = P_{ref}$, and r_y is chosen such that $C_{ry} = P_{ref}$. The SL between X and Y can now be formally defined as:

$$SL=\frac{2}{N(N-\omega)P_{ref}}\sum_{i=1}^{N}\sum_{j=i+w}^{N-\omega}[\theta(r_x-|X_i-X_j|)\theta(r_y-|Y_i-Y_j|)] \quad (3)$$

SL is a symmetric measure of the strength of synchronization between X_i and Y_i. In the equation (3), the average is calculated over all i and j; by doing the averaging only over j, SL can be computed as a function of time *i*. From equation (3), it can be seen that in the case of complete synchronization SL=1; in the case of complete independence SL=P_{ref}. In the case of intermediate levels of synchronization $P_{ref}<SL<1$.

B. *Applications running on Grid*

The deployment of an efficient and reliable GRID application is far from being trivial: it requires a first phase of algorithmic and data analysis, in order to choose the best paradigm matching the application. Later, these notions need to be translated into parallel code. In this section, the approach and techniques to meet the goal of a smooth and efficient migration of the Matlab application described in this paper, are presented.

First of all, the Matlab project needs to be translated into machine code, in order to be executed successfully over the Unix GRID environment. Since worker nodes cannot include a full working version of Matlab, Mathworks provides its own full-fledged compiler, called mc. The latter is employed to create a stand-alone binary executable of the Matlab code that is spread across the nodes for further running.

Due to the complexity of the computations, we chose to use a divide and conquer approach: the source dataset is recursively divided to simplify the original problem, without losing any generality and/or precision. Using a special crafted bash script, we split the datasets into small chunks to balance the load of every node involved in the simulation. For example, if the participating nodes are a set of 120 worker nodes, the original dataset is divided into 120 small ones, too. Since each dataset is a huge amount of data, they are successively moved to remote storage elements, using a special tag able to identify the worker nodes responsible for their computation. Distributed applications running over several machines can use various methods to exchange data between nodes and processes. Security and confidential constraints impose us to use ciphered connections: the natural choice is to use LCG file transfer API, built-in in our gLite framework.

Job behaviors are monitored and controlled using the DAG-Jobs Grid extension. A DAG (Directed Acyclic Graphs) represents a set of jobs in which input, output and execution of one or more of them, depends on one or more jobs. Edges (arcs) represent the dependencies while jobs are nodes (vertices) in the graph. There can be a set of common attributes valid for all jobs: in our framework, the shared attribute among child jobs is the Matlab compiled code, generated as described above. After the submission, the DAG collection is validated against credentials and resource requirements. If conditions are met, every job fetches its own data chunk from storage elements, in order to run its work. After the completion, every node sends a signal indicating the "done" status. When all child jobs have been completed, the parent one retrieves all outputs and merges them. The result is checked against possible errors: if no errors are discovered, the user is notified about the completion of the work; otherwise, he is prompted to launch again the DAG. The entire process is continuously monitored using the gLite framework built-in monitor functions. We used the glite-wms-jobs-status family functions to get easy job status information from the book-keeping service (i.e. LB

service), available in CO.M.E.T.A. GRID infrastructure. Finally, the user is asked to delete the source data file to free the previously used space in the storage element. A divide-et-impera paradigm allow us to speed up computations about ten times: the same dataset, running on an Intel Core2Duo at 2.66Ghz, took about twenty hours to complete, against only two hours needed by the GRID version. In future works, further optimizations using the Matlab *parfor* paradigm will allow us to make thin the GRID control code. Moreover, we expect an overall job improvement of about eighteen percent, due to MPI libraries used.

C. Network Theory

The matrix was built using pair wise SL values for each couplet of head channels, by using a virtual platform [27] for biopotentials study, n was converted to an un-weighted graph where the nodes represent the channels and where, by applying a specific threshold, an edge was created between those nodes having a SL value above the threshold, and no edge in the case of a sub-threshold SL. The resulting graphs can be easily characterized through parameters such as the clustering coefficient $Cp^{(i)}$, the characteristic path length $Lp^{(i)}$ and the degree $k^{(i)}$ calculated for the node i^{th} and then extended to the entire network as *Cp*, *Lp* and *k*, averaging them over all nodes.

In particular, the clustering coefficient is defined as the measure of the local interconnection of the graph, whereas the path length is an indicator of its overall connection. Watts and Strogatz (1998) have shown that graphs with many local connections and a few random long distance connections are characterized by a high cluster coefficient and a short path length; the networks having these properties are known as ''small-world''.

With a probability p, a random edge is chosen and rewired to be connected to a randomly chosen vertex. By varying p between 0 and 1, different graphs can be created which span the whole range from regular (p = 0) to random (p= 1).

Regular networks of N nodes have a high *Cp* ($Cp \approx 3/4$) but a long characteristic path length ($Lp \approx N/2k$); random graphs have a low *Cp* ($Cp \approx k/N$) but the shortest possible path length ($Lp < \ln(N)/\ln(k)$). Thus, the discovery of Watts and Strogatz proves that networks with $0 < p < 1$ have a path length that is much smaller than that of a regular network, while the *Cp* is still close to that of a regular network.

Assuming that :

- $\gamma = C_{network} / C_{rand}$
- $\lambda = L_{network} / L_{rand}$
- $\sigma = \gamma / \lambda$

the small-world networks are characterized by having $\gamma > 1$ and $\lambda \approx 1$, thus $\sigma \geq 1$.

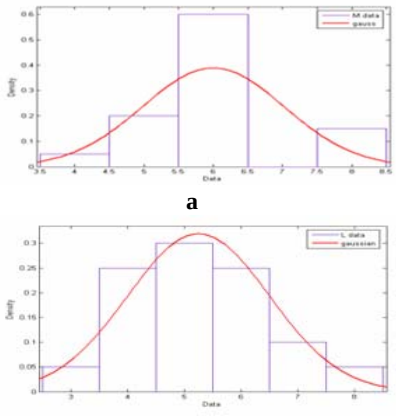
Many types of real networks have been shown to have small-world features [10]. Moreover, different patterns of anatomical connectivity in neuronal networks show a high clustering and a small path length [18].

IV. RESULTS

Methods applied in this study investigate the occurrence of small-world phenomenon in MEG data by considering the application of the SL method and the characterization of the respective graphs obtained by varying the threshold T.

The first step is the characterization of the parameters needed for the SL method. The P_{ref} value has been set equal to 0.01, and the Theiler correction *w* has been considered null, to be comparable with the SL method application in other studies [2].

The embedding parameters have been evaluated using a software specific for nonlinear time series analysis, TISEAN [20]. The distributions density of the time lag *L* and of the embedding dimension *m,* evaluated for each signal, has been executed in order to choose the same values for each time-series.



a

b

Figure 2 *Distribution of (a) Time Lag (L) and (b) embedding dimension m.*

According to the *L* and *m* distributions shown in Fig.2, the time lag has been set equal to *L*=5 and the embedding dimension equal to *m*= 6 for all time series.

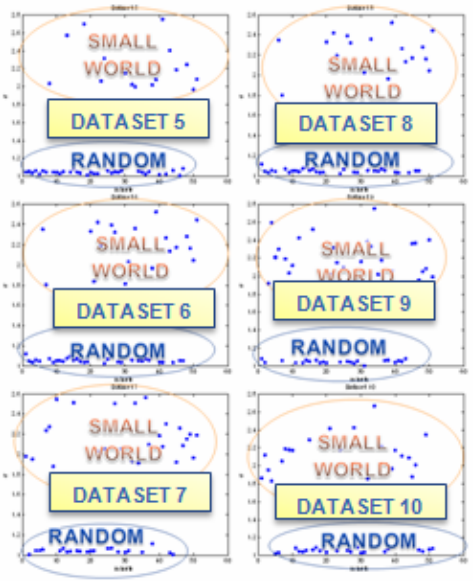


Figure 3 *Network topologies. X-axes represent the minute of analysis and in the y-axis the value of σ is reported.*

In the next step the SL matrices have been calculated for each minute on GRID architecture. The synchronization matrix is a square matrix where the x axis and the y axis correspond with the channel numbers, and where the entries indicate the strength of the SL between specific pairs of channels. The matrices of pair wise SL values were converted to un-weighted graphs by assuming an edge between pairs of channels with a SL above a threshold *T*, and no edge in the case of a sub-threshold SL *T* and were studied by using networks parameters K, L and C. The topologies of the networks have been investigated calculating the σ values for each minute, and small world architectures have been found in each dataset as shown in Figure3.

V. Conclusions

The effort here was directed towards an understanding of the neurophysiologic processes exhibited by MEG signals.

The aim of this work was to investigate the possibility of finding small-world architectures in functional connectivity with MEG data, and in examining the architecture from the data analyses. The method used to examine the statistical interdependencies among these signals, Synchronization Likelihood (SL), was time consuming. Therefore, distributed computing turned out to be a powerful instrument to help speed up simulations and process the very large quantities of data.

The SL method has been combined with network theory, after the GRID SL matrices calculation. The matrices of pair wise SL values were converted to un-weighted graphs and they have been studied and characterized by three parameters: the clustering coefficient, the shortest path length, and the degree of distribution. Furthermore, the possibility of extracting network architectures that can represent the functional connectivity in MEG data, or in other spatially extended biopotentials, is an important first step for a more extensive study in which the characterization of these networks, using their parameters (*Cp*, *Lp* and *k*), can be used for the discrimination of different topologies or as a

diagnostic. A statistical analysis can be done by exploring the variation of SL matrices by varying the parameters as P_{ref}, w and m and L.

In addition, some embedded Matlab features for parallel processing can be adopted to minimize the computing time.

REFERENCES

[1] Kandel ER, Schwartz JH, Jessell ThM: Principles of neural science. *McGraw-Hill*; 2000.

[2] Stam CJ, Jones BF, Nolte G, Breakspear M, Scheltens Ph: "Small-world networks and functional connectivity in Alzheimer's disease". Cereb Cortex 2007, 17:92-99.

[3] Le van Quyen M: "Disentangling the dynamic core: a research program for a neurodynamics at the large scale". *Biol Res* 2003, 36:67-88.

[4] Stam CJ: Nonlinear brain dynamics New York: Nova Science Publishers;2006.

[5] Linkenkaer-Hansen K, Nikouline VV, Palva JM, Ilmoniemi RJ: "Longrange temporal correlations and scaling behavior in human brain oscillations". *J Neurosci* 2001, 21:1370-1377.

[6] Watts DJ, Strogatz SH: "Collective dynamics of "small-world"networks". *Nature* 1998, 393:440-442.

[7] Amaral LAN, Ottino JM: "Complex networks. Augmenting the framework for the study of complex systems". *Eur Phys J B* 2004, 38:147-162.

[8] Lee L, Harrison LM, Mechelli A. 2003. "A report of the functional connectivity workshop, Dusseldorf 2002". *Neuroimage* 19:457--465.

[9] Stam CJ, Van Dijk BW. 2002. "Synchronization likelihood: an unbiased measure of generalized synchronization in multivariate data sets". *Physica D* 163:236--241.

[10] S.H. Strogatz, "Exploring complex networks", *Nature* 410 (2001) 268–276.

[11] L.F. Lago-Fernandez, R. Huerta, F. Corbacho, J.A. Siguenza, "Fast response and temporal coherent oscillations in small-world networks", *Phys. Rev. Lett.* 84 (2000) 2758–2761.

[12] Cornelis J Stam , Jaap C Reijneveld, Graph theoretical analysis of complex networks in the brain. Nonlinear Biomed Phys. 2007 ;1 (1):3 17908336

[13] O. Sporns, G. Tononi, G.E. Edelman, "Connectivity and complexity: the relationship between neuroanatomy and brain dynamics", Neural Netw. 13 (2000) 909–922.

[14] F. Sapuppo, E. Umana, M. Frasca, M. La Rosa, D. Shannahoff-Khalsa, L. Fortuna and M. Bucolo, 'Complex Spatio-Temporal Feature in MEG Data', Mathematical Biosciences and Engineering, October 2006, Vol. 3, No. 4, pp. 697-716.

[15] Motter AE, Matias MA, Kurths J, Ott E: Dynamics on complex networks and applications. Physica D 2006, 224:vii-viii.

[16] Bassett DS, Meyer-Linderberg A, Achard S, Duke Th, Bullmore E: Adaptive reconfiguration of fractal small-world human brain functional networks. PNAS 2006, 103:19518-19523.

[17] Takens F. 1981. "Detecting strange attractors in turbulence". In: Morel JM, Takens F, Teissler B, editors. Lecture Notes in Mathematics. Volume 898. New York: Springer-Verlag. p 366--381.

[18] Strogatz SH. 2001. Exploring complex networks. Nature 410:268--276.

[19] Theiler J. 1986. Spurious dimension from correlation algorithms applied to limited time-series data. Phys Rev A 34:2427--2432.

[20] R. Hegger, H. Kantz, and T. Schreiber, Practical implementation of nonlinear time series methods: The TISEAN package, CHAOS 9, 413 (1999).

[21] P.L. Nunez, R. Srinivasan, A.F. Westdorp, R.S. Wijesinghe, D.M. Tucker, R.B. Silberstein, P.J. Cadusch, EEG coherency. I. Statistics, reference electrode, volume conduction, Laplacians, cortical imaging, and interpretation at multiple scales, Electroenceph. Clin. Neurophysiol. 103 (1997) 499.

[22] P.L. Nunez, R.B. Silberstein, Z. Shi, M.R. Carpenter, R. Srinivasan, D.M. Tucker, S.M. Doran, P.J. Cadusch, R.S. Wijesinghe, EEG coherency. II. Experimental comparisons of multiple measures, Clin. Neurophysiol. 110 (1999) 469.

[23] Franaszczuk, P. J. and Bergey, G. K. An autoregressive method for the measurement of synchronization of interictal and ictal EEG signals. Biol Cybern 1999; 81:39.

[24] M.G. Rosenblum, J. Kurths, Analysing synchronization phenomena from bivariate data by means of the Hilbert transform, in: H. Kantz, J. Kurths, G. Mayer-Kress (Eds.),Non-linear Analysis of Physiological Data, Springer Series for Synergetics, Springer, Berlin, 1998, pp. 91–100.

[25] Shannahoff-Khalsa, DS, Selective Unilateral Autonomic Activation: Implications for Psychiatry (a review article), CNS Spectrums: The International Journal of Neuropsychiatric Medicine, 12(8), pp. 625-634, Aug 2007.

[26] Shannahoff-Khalsa DS, Ray LE, Levine S, Gallen CC, Schwartz BJ, Sidorowich JJ. Randomized Controlled Trial of Yogic Meditation Techniques for Patients with Obsessive Compulsive Disorders, CNS Spectrums: The International Journal of Neuropsychiatric Medicine, vol 4, no. 12, pp 34-46, 1999.

[27] M. Bucolo, F. Di Grazia, M. Frasca, L. Fortuna, F. Sapuppo, 'BioS: a New Tool for Biopotential Experiments', *The 29rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society' (EMBC 2007),* Lyon, France, 23-26 Agosto 2007, pp. 1969- 1972.

[28] Hilgetag CC, Burns GAPC, O'Neill MAO, Scannell JW, Young MP: Anatomical connectivity defines the organization of clusters of cortical areas in the macaque monkey and the cat. Phil Trans R Soc Lond B 2000, 355(1393):91-110.

Federica Di Grazia received the Electronic, Automation and Complex Systems Engineering Master degree from the University of Catania in 2007. She is a PhD Student in Electronics, Automation and Control of Complex Systems, University of Catania, from November 2006. Her research activity mainly regards nonlinear time series analysis, complex system theory and diagnostic support system for the analysis of physiologic bio-potential.

Francesca Sapuppo currently conducts research, as a postdoctoral associate, at the DIIES, University of Catania and she is visiting professor at the Department of Electronic Engineering at the University of Messina. She received the M.S. degree in microelectronics engineering in 2003 and Ph.D. degree in electronic and control engineering in 2007 all from the University of Catania, Italy. Currently, her research includes bio-microfluidic control and system engineering, nonlinear technologies as cellular neural networks (CNN) applied to distributed real time monitoring and control of microfluidic processes. Her research includes also nonlinear data analysis and chaos theory applied to biological and microfluidic phenomena.

David Shannahoff-Khalsa is the Director of The Research Group for Mind-Body Dynamics, Institute for Nonlinear Science, University of California, San Diego. He formed this interdisciplinary inter-institute research group with colleagues in 1994. Prior to that he was at The Salk Institute for Biological Studies in La Jolla, California, from 1970 through 1993. His original studies were in nucleic acid chemistry and biochemistry. In 1976 he changed fields to study concepts and techniques in yogic medicine. His work has led to new concepts for understanding psychophysiological states via studies employing multi-variate physiology of the central and autonomic nervous systems and the other major bodily systems in humans. He has also pioneered work on the brain effects of specific yogic meditation techniques and how they can be applied for treating psychiatric disorders. One of his research projects is as a PI on a grant in Brazil where yogic meditation techniques will be used to help prevent and treat psychiatric disorders in children and adolescents.

Luigi Fortuna is a full professor of system theory at the University of Catania, where he is dean of the Engineering Faculty and coordinator of the Ph.D. course in Electronics, Automation and Control of Complex Systems. He is author of more than 400 scientific publications. He is author of ten industrial patents. He is in charge of contracts with public and private companies, such as exMURST, CNR, ENEA, EURATOM, ERG Petroli, ASI, and STMicroelectronics. He is responsible for the FIRB project, nationally—" Cellular self-organizing nets and chaotic nonlinear dynamics to model and control complex systems." He is an IEEE Fellow, and chair of the IEEE Committee on CNNs.

Maide Bucolo received the Computer Science Engineering degree from the University of Catania in 1997 and the Ph.D. degree in Electronic and Control Engineering in 2001. She is currently Assistant Professor at the University of Catania. Moreover she is involved in several projects supported by public institutions and consortium of private companies. Her research activity mainly regards study of complex dynamics, cellular neural networks and nonlinear time series analysis for biomedical and economic applications. She is *IEEE* member.

PhD **Costantino Pistagna** was born in Catania on 13th March, 1977. He got his graduation in Computer Science at the University of Catania on July 2003. After graduation he hold a Ph.D. from university of Catania about peer-to-peer networking and overlay networks. He worked as consultant for important telecommunication societies in Italy, like Fastweb and Telecom Italia. Since April 2007, he is involved in the COMETA project, as a member of Application Identification and Support Team and Hardware and Software Technical Support Team. He took part, as trainer, in many induction and dissemination events in Italy. His actual interests include networking and peer-2-peer system simulations and modeling, genetic algorithms and game theory as solving tools for networking problems. He is currently working as system administrator of the Grid-node of Department of Mathematics and Computer Science (University of Catania).

Dose Distribution Verifications in Medical Physics: Results Obtained Using Geant4 and Grid Technology

V. Ardizzone[1], R. Barbera[1], G. A. P. Cirrone[2], G. Cuttone[2] , F. Di Rosa[2], R, Foroni[3], E. Giorgio[1], G. La Rocca[1], S. Lo Nigro[4], F. Romano[4], G. Russo[5], M. Russo[6], M. G. Sabini[7]

[1]*Istituto Nazionale Fisica Nucleare, Section of Catania, Catania, Italy*
[2]*Istituto Nazionale di Fisica Nucleare, Laboratori Nazionali del Sud, Catania, Italy*
[3]*Gamma Knife Center, Borgo Trento Hospital, Verona, Italy*
[4]*Department of Physics, University of Catania, Catania, Italy*
[5]*Fondazione Istituto San Raffaele - G. Giglio di Cefalù, Cefalù (PA), Italy*
[6]*Consorzio Cometa, Catania, Italy*
[7]*Department of Imaging, Cannizzaro Hospital, Catania, Italy*

***Abstract*— In the last years our research group has been involved in the development of Monte Carlo simulations in the medical field for the calculation of dose distribution curves. Accuracy is of great importance for these applications where a huge number of events is necessary to decrease the statistical uncertainty. This results in long calculation times which the traditional clusters cannot sensibly decrease. The use of Grid technology has permitted the calculation of dose distributions, in different and complex configurations, drastically reducing the calculation time and maintaining the same level of accuracy. In this work we show two different examples of Monte Carlo simulation realized with the Geant4 toolkit, both implemented in the Grid environment: Hadrontherapy and GammaKnifeRS. Hadrontherapy is an application which simulates the proton therapy beam line operating at the Laboratori Nazionali del Sud of the INFN in Catania, Italy. Each element of the beam line has been detailed reproduced and a validation against experimental data has been carried out. GammaKnifeRS is an application which simulates the Elekta Gamma Knife unit, a radio-surgical device operating at the Cannizzaro Hospital in Catania. Also in this case the complex geometry of the collimators and radioactive sources has been reproduced in detail but a bigger number of events was necessary. We used up to about 100 nodes and a "N-out-of-M" strategy was exploited. It consists on scheduling more subtasks than really required in order to avoid that a slower node becomes the bottleneck of the whole computation. For both the simulations, the production testbed available within the COMETA Consortium was exploited and a number of nodes ranging from 20 up to 120 has been used. Our work shows that the Grid technology can represent in general a solution for long calculation time required by the Monte Carlo method, especially in the medical field where accuracy and reasonable time are two unavoidable constraints.**

***Index Terms*— Monte Carlo, Grid, proton therapy, radiosurgery**

I. INTRODUCTION

In recent years, the sophistication and complexity of clinical treatment planning systems have increased significantly, especially for three-dimensional (3D) treatment planning systems, the delivery techniques, and the use of conformal treatment planning. In particular an accurate verification of dose calculation must be employed for the *high conformational* techniques like proton therapy and stereotactic radio-surgical. In both cases, in fact, high doses

are delivered near critical anatomical structures.

The Monte Carlo approach, on the other hand, can be considered, nowadays, one of the most accurate methods for released dose calculation in tissue. These considerations lead up our Group to develop two different Monte Carlo applications: *Hadrontherapy* and *GammaKnifeRS*. They were developed using the Geant4 (Geometry ANd Tracking) [1][2] toolkit, a powerful C++ based set of libraries, firstly developed at CERN for High Energy Physics and nowadays also applicable in the low-energy range, according to the medical concern.

The realized applications are able to fully simulate a typical proton therapy beam line and a Gamma Knife device. *Hadrontherapy* and *GammaKnifeRS* have been then ported in a Grid environment, this permitting to exceed the major drawback of Monte Carlo method, i.e. the long computation times.

In this work we will report on the results obtained in the application and we will show the power of Grid approach for Monte Carlo based clinical applications.

A. The Hadrontherapy application

Hadrontherapy [3] is a Geant4 example able to precisely reproduce all the elements of a generic proton-therapy transport beam line. The line can be reconstructed from the point in which beam exits in air up to the isocenter (the point in which the patient is positioned during the irradiation). Such elements are the diffuser, the monitor chambers, the range shifter, the energy modulation system and the various collimators; the geometrical and physical characteristics of these elements can be easily defined and changed. Moreover it is possible to choice the characteristics of the incident beam: beam energy, energy spread, beam spot size and angular spread. We simulated two different detectors, a Markus ionisation chamber and a generic radiochromic film, in order to study the depth dose deposition curves (Bragg curves) and lateral dose distributions inside a phantom.

In a proton therapy treatment the doctor and the medical physicist choose, for each patient, a specific beam line configuration. This configuration includes a patient-specific range shifter (to degrade the energy beam), range modulator (to modulate in energy the beam) and a final collimator. The final collimator and the modulator wheel are realized on the basis of the information received from the treatment planning system. The angular aperture of modulator is related to the specific thickness of the tumour while the collimator shape coincides with the specific tumour section.

The example Hadrontherapy can be an important instrument for all the centres are developing a new proton therapy facility and are designing and optimising the transport beam line and/or check the dose distributions before their experimental measurements. It can also represent a preliminary step towards the development of a Monte Carlo treatment planning system, based on the GEANT4 toolkit.

The application, at the end of each simulation run, furnishes to the user a volumetric matrix containing information on the 3D dose distributions deposited in a given material with a spatial resolution up to10 μm. This kind of information can be used by the radiotherapist and medical physicist to reconstruct all the dose distribution to plan a treatment.

One of the main disadvantages of Hadrontherapy is the long computation time required for a complete and statistically reasonable, simulation. On the other hand, fast responses are needed for clinical applications where medical doctors and patients cannot wait long times before start with the treatment.

In order to find a solution for a concrete application of Monte Carlo technology in the clinical practice, we decided to adopt the Grid technology. Grid permits to split a single big job in many smaller jobs reducing exponentially the computation times. In our work we will demonstrate that the computation times, necessary for a typical proton therapy simulation based on the use of GEANT4, can decrease approximately of a factor 20 using 20

CPUs INTEL-Xeon at 2.8 GHz. Moreover we realized a very simple web page in which users can insert the main inputs of their simulation (beam energy, number of jobs, number of events, etc.) and directly collect the results without interact with the C++ code of the application. The application realized has been tested simulating the CATANA proton beam line and comparing the results with the experimental data.

1) Implementation of Hadrontherapy on the Grid

One of the main limitations in the use of the Monte Carlo simulations in medical applications is the long computation time necessary to obtain a good statistical result. In a Pentium 4 2 GHz CPU, the run time of our application, to track 1 million of protons, is 4.16 hours: absolutely not compatible for a medical application to be used in a real clinical case. One possible solution, to obtain a reduction of the computation times, is the distribution of the jobs on many CPU's using a computing system like the Grid. For our tests we installed the GEANT4 application on the local Catania Grid worked node. The first step to run our application on Grid was the building of the RPM4 of GEANT4 toolkit. This step is necessary in order to configure in the same manner all the computers of the Grid. Moreover the RPM of Geant4 can be used from GEANT4 community to simplify the installation of the entire package. The second step consisted in the realisation of some Linux shell scripts: these permit to launch different jobs at the same time, (it is possible to define the number of jobs to launch), and to get output files, to check the status of jobs and to get the output files at the end of the simulation. The final step was the implementation of a simple, user friendly, web interface inside the Genius portal. It permits the interaction with the simulation in a very transparent way: the user can send his jobs just clicking on a button and the interface will start the simulation, split the jobs over the various CPUs, collect the final results and, finally, plot the desired distributions.

B. Some result obtained with Hadrontherapy

Once the Grid implementation was completed, our first action was to compare the Bragg curve obtained in one computer with that obtained running Hadrontherapy on 10 computers. The total number of events was the same. In order to quantify the agreement between the two distributions, the test of Kolmogorov-Smirnov (K-S) in the Goodmann approximation, was performed. Confidence level is fixed at 5% for the rejection of the null hypothesis. From a statistical point of view the K-S test is more powerful than the simple χ^2 test. Comparing the Bragg curve distributions (calculated in one computer and distributing the jobs) using the K-S test, we found an *equivalent* χ^2 probability of 0.99. Figure 1 shows the comparison of the two Bragg curves. We studied the total computation time versus the number of CPUs used.

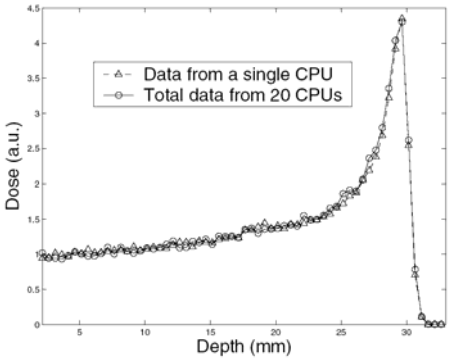


Fig. 1. Comparison between a depth dose curve obtained from one CPU and that obtained distributing the calculation in 20 CPUs. The number of total histories is constant in the two cases.

It decreases exponentially: from 4.16 hours, for one million histories, down to 15 minutes distributing the calculation in 20 machines. From the obtained results we found that about 85 CPUs are necessary to decrease the computation time, to obtain a statistically reasonable simulation, down to 3 min. A such time could be acceptable also for a routinely

use of Monte Carlo. In order to simplify the interface between the user and the Grid-GEANT4 application, a web portal was created. It permits to set the simulation parameters (energy of the beam, position of the beam-line elements, etc.), to define the number of events and the number of jobs to be submitted, to download the simulation output (given by the sum of the single job results) and to display, in form of plot, the final result. Figure 2 shows the web portal realized after the run of a simulation. The resulted Bragg curve is also displayed.

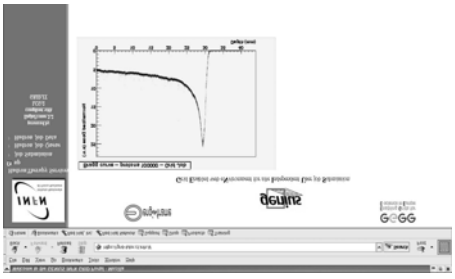


Fig. 2. The developed web interface for the use of the Grid with Hadrontherapy. The output esult from one simulation is shown.

C. The GammaKnifeRS application

The Leksell Gamma Knife is a device for stereotactic radiosurgery. It is used to treat different brain disorders which are often inaccessible for conventional surgery, such as benign or malignant tumors, arteriovenous malformations and trigeminal neuralgia. This devices employs a Treatment Planning System (TPS) based on a simple algorithm, whose name is Leksell GammaPlan (LGP). It calculates the dose delivered to a water phantom according to the image data of the patient (TC or MRI or angiography) [4]. In order to compute the dose distribution in a reasonable time, LGP employs a semi-empirical algorithm based on pre- measured beam dose profiles and it assumes the patient's head composed of water. These approximations can achieve uncertainties in the dose computation especially for particular configurations, such as in presence of different density materials. The difficulties inherent to measurements, above all for small beams, make Monte Carlo method a valuable and powerful alternate tool, now clearly recognized by medical physics community to validate the TPS dose distribution also in particular configurations. In literature several studies can be found where results of LGP and Monte Carlo calculation are compared using different simulation toolkits, such as EGS4[5], PENELOPE [6] and MNCP [7]. In these papers results are provided both for homogeneous and inhomogeneous phantoms, simulating at first one source and then the multi-source configuration.

The aim of our work has been to explore the Grid technology capability to reduce the Monte Carlo time run necessary for validating the LGP TPS according to the international guidelines for a clinical case. In particular we have developed the application GammaKnifeRS (Gamma Knife Stereotactic Radiosurgery) by using Geant4 (ver. 8.2 patch-01), for the simulation of a. Each component of the Gamma Knife unit has been simulated in details. The simulation of the complex geometry and of the different radiation configurations used in the clinical application need a long computation time run to achieve the accuracy demanded. In order to decrease it, we exploited the Grid technology, using the production testbed available within the COMETA consortium.

In this section of the paper we will explain the procedures used to implement the Grid technology into a Geant4 Monte Carlo simulation and our experiences.

1) Short description of the simulation code

Details about the geometry of the Gamma Knife beams have been obtained from the manufacturer's technical documentation. The Gamma Knife consists of an iron body containing 201 ^{60}Co sources in a hemispherical arrangement. Gamma ray beams converge

through a collimator system to a common focal point, called isocentre, where the target volume has to be positioned. By combining different focal points it is possible to cover complex shaped targets, with minimum impact on the surrounding healthy tissues. This multi-shots treatment is possible thanks to a precise Automatic Position System (APS), connected to a coordinate frame fixed to the patient's head. The collimation system is composed of stationary collimators and a mobile helmet; the final beam size is achieved by using different available helmets (4, 8, 14 and 18 mm) with an appropriate configuration of plugged sources [8]. At the isocentre was placed the centre of a spherical water phantom in order to simulate the patient's head (Fig. 3). The phantom has the same size and material of that used for the measurements. A cube of 45 mm of side was placed in the centre to compute the total dose absorbed at the end of the simulation (Fig. 4). All the most important physical processes

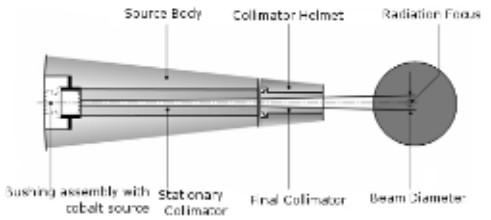


Fig. 3. Simple scheme of one of the 201 sources with the respective collimation system, which determines the beam shape inside the spherical phantom.

concerning the energy range of interest were included both for primary and secondary particles. A clinical radio-surgical treatment of a cavernous sinus meningioma has been simulated, it is composed by seven shots exploiting two different collimators. A total number of $7 \cdot 10^9$ primary particles were sampled to achieve an accuracy of 2%. The time run needed in our available cluster, composed of 20 Pentium 4 CPU, was too large to obtain the result in a reasonable time. So we have employed the Grid technology.

2) Monte Carlo parallelization

Parallelization is a way to accelerate the convergence of a Monte Carlo computation. If N processors execute N independent copies of a Monte Carlo computation the accumulated result will have a variance N time smaller than that of a single copy. In a distributed Monte Carlo application, once a distributed task starts, it can usually be executed independently with almost no inter-process communication. In this case a large task is split into smaller independent subtasks and each of them is then executed separately. This method requires that the random seed in each subtask have to be independent in a statistical sense (Fig. 5). A simple way to do that automatically is to set the seed of each simulation as a function of the CPU time. This method was exploited here and moreover each job has been sent with ten seconds of delay, in order to avoid the same CPU time in different subtasks. A bash script

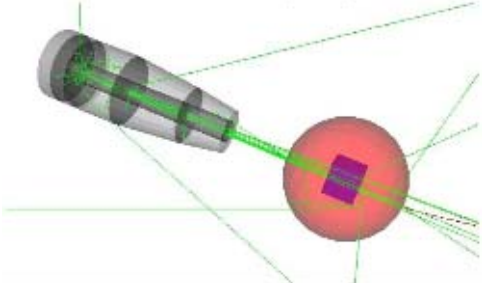


Fig. 4. Simulation of the Gamma Knife as it appears in Geant4: collimation system in grey, water phantom in red and scoring cube in blue. Trajectories of photons (green) and electrons (red) are also visible.

divides the total histories of the Monte Carlo application in many subtasks and it sends these to the Grid. The user can choose the number of subtasks and the total histories to send.

When the subtasks are done, the collection service can be used to gather the results and generate the final result of the entire computation. In our case we have developed a Matlab application to sum and analyze the results automatically.

3) Grid implementation

The characteristics of Monte Carlo applications motivate the use of Grid computing to perform large-scale Monte Carlo computations, like in our medical application. The Grid implementation for a GEANT4 Monte Carlo application does not need to modify any code line so it is very friendly for the user, who only has to install the GEANT4 toolkit libraries on each node of the Grid. This operation can be executed automatically and easily by using some Grid commands, so the user only has to make the tarball of the GEANT4 libraries. The Grid characteristics are different in respect to conventional clusters and for this reason the splitting method has to be different. The nodes that provide CPU in a Grid system will most likely have computational capabilities that vary greatly. A node might be a high-end supercomputer, or a low-end personal computer, even just an intelligent widget. In addition, these nodes are geographically distributed and not centrally manageable: a node may go down or become the bottleneck of the whole computational system if the assembly of the final result must wait for the partial result generated on this slow node. Moreover a halted subtask might prevent the whole task from ever finishing. However, the statistical nature of Monte Carlo applications provides a shortcut to solve this problem at the application level.

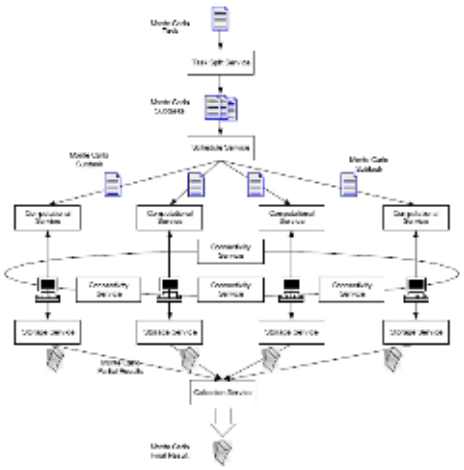


Fig. 5. Scheme of a Monte Carlo application splitting on a Grid system.

Suppose we are going to execute a Monte Carlo computation on a Grid system. We split it into N subtask, with each subtask based on its unique independent random number stream. We then schedule each subtask onto the nodes in the Grid system. Of course, the assembly of the final result requires all the N partial results generated from N subtasks. Each subtask is a ”key” subtask, since the suspension or delay of someone of these will have a direct effect on the completion time of the whole task, as showed in the example in Figure 6. When we are running Monte Carlo applications, what we really care about is how many random samples (random trajectories) we must obtain to achieve a certain, predetermined, accuracy. It is not relevant which random sample set is used, provided that all the random samples are independent in a statistical sense. The statistical nature of Monte Carlo applications allows us to enlarge the actual size of the computation by increasing the number of subtask from N to M, where M > N . Each of

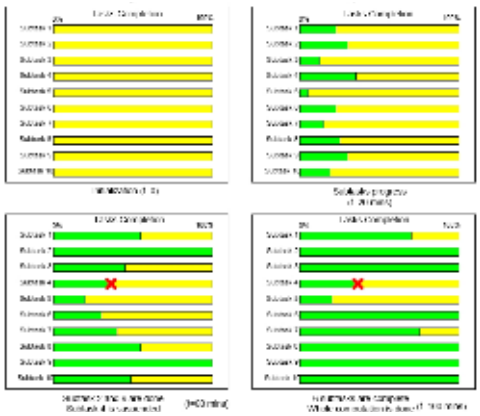


Fig. 6. Example of the ”6-out-of-10” Strategy with 1 Suspended and 3 ”Slow” Subtasks.

these M subtasks uses its unique independent random number set, and we submit M instead of N subtasks to the Grid system. Therefore, M bags of computation will be carried out and M partial results may be eventually generated. However, it is not necessary to wait for all M subtasks. When N partial results are ready, we consider the whole task for Grid system completed. The application then collects the N partial results and produces the final result. At this point, the Grid-computing system may broadcast abort signals to the nodes that are still computing the remaining subtasks. So we can tolerate at most M-N delayed or halted subtasks [9]. This method is called the N-out-of-M.

4) The N-out-of-M method

In Monte Carlo applications, N is determined by the ap- plication and it depends on the number of random samples or random trajectories needed to obtain a predetermined accuracy. The problem is thus how to choose the value M properly. A good choice of M can prevent from delaying or even halting the whole computation. However, if M is chosen too large, there may be little benefit to the computation at the cost of significantly increasing the workload of the Grid system. In order to determine a proper value of M, we have studied the Grid behavior and we have considered some system parame- ters. In the N-out-of-M strategy, the completion time of the Monte Carlo computational task depends on the performance of each individual node that is assigned to a subtask, the node failure rate, and also the interconnection network failure rate. For our work we have exploited a homogeneous Grid system composed by 300 CPU AMD Opteron 2218 with 8 GB RAM linked together by an ”Infiniband 4X”. This system uses resources of the ”PI2S2 Project” managed by the COMETA consortium, a project co-funded by the Italian Ministry of University and Research (MIUR) [10]. In this case only the scheduled rate of subtask in a single node has been considered because the node and the interconnection network failure rate are negligible. However, if we increase M too much, the workload of the system increases without significantly reducing the Monte Carlo task completion time. Therefore, we can expect that, with a proper number of subtask scheduled using the N-out-of-M strategy, the Monte Carlo completion time on a Grid can be made to be the same as the subtask completion time in a node. Really, the Monte Carlo completion time is often bigger because all subtasks do not start at the same time since the number of free nodes is minor than the number of needed nodes, M. In this case the value of M is similar to N because the Grid system used is homogeneous and reliable. To obtain the accuracy needed the minimum number of subtask, N, is 336, each samples 20 million of histories, so we have chosen M equal to 350. Each subtask has a completion time of about 10 hours, but the whole task has a mean completion time of 16 hours because all subtask do not start at the same time since the number of free nodes is minor than number of sent subtask. However, the time needed is so small to achieve our work successfully.

II. Conclusion

The sending of any GEANT4 application on Grid is easy and it does not need any code modification. Moreover the Grid technology is a good solution to reduce the Monte Carlo completion time for quality assurance studies. We used a advantageous method called N-out-of-M in order to optimize the scheduling and completion time. So the Grid could be a good solution to clinical dose distribution validation needed in several radiotherapy treatments.

Acknowledgments

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian

Ministry of University and Research (MIUR) within the Programme Operativo Nazionale ”Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione” (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it. The authors gratefully acknowledge Dr. Gianmarco De Francisci Morales and Dr. Simona Blandino of the Department of Computer and Telecommunications Engineering University of Catania and Consorzio COMETA for their technical support.

REFERENCES

[1] S. Agostinelli et al.,Geant4 - a simulation toolkit, Nucl. Instr. Meth. in Phys. Res. A 506, 250-303(2003).
[2] J. Allison et al., Geant4 applications and developments, Trans. Nucl. Sci., 53, 270-278 (2006).
[3] G.A.P. Cirrone et al.,The GEANT4 toolkit capability in the hadrontherapy field: simulation of a trasport beam line, Nucl. Phys. B - Proc. Suppl. 150 (2006) 54.
[4] Elekta, Reference Manual for Leksell GammaPlan, Stockholm: Elekta Instruments, 2003.
[5] J.Y.C. Cheung et al., Monte Carlo calculation of a single-beam dose profiles used in a gamma knife treatment planning system, Med. Phys 25(9), 1673-1675 (1998)
[6] V. Moskvin et al., Monte Carlo simulation of the Leksell Gamma Knife: I. Source modelling and calculation in homogeneous media, Phys. Med. Biol. 47, 1995-2011(2002)
[7] J. Trnka et al., MNCP-based computational model for the Leksell Gamma Knife, Med. Phys. 34(1), 63-75(2006)
[8] Elekta 2003, Leksell Gamma Unit - User’s Manual (Stockholm: Elekta Instrument).
[9] Yaohang Li and Michael Mascagni, Analysis of Large-scale Grid-based Monte Carlo Applications, Departement of Computer Science and School of computational Science and Information Technology, Florida State University.
[10] COMETA website, www.consorzio-cometa.it, www.pi2s2.it.

Porting of a Docking Application on PI2S2 Grid infrastructure

A. Pedretti[1], A. Lombardo[2,3], G. Vistoli[1], E. Mastriani[3], F. Pappalardo[4] and S. Motta[4]

[1]Dipartimento di Scienze Farmaceutiche "Pietro Pratesi", Università degli Studi di Milano, Via L. Mangiagalli 25, 20133 Milano, alessandro.pedretti@unimi.it
[2]Parco Scientifico e Tecnologico della Sicilia, Z.I. Blocco Palma I, Stradale V. Lancia 57 95121, Catania, alombardo@pstsicilia.org
[3]Consorzio Cometa, Via S. Sofia 64,I-95123 Catania
[4]Università di Catania

***Abstract*— Drug discovery is an extended process that can take as many as 15 years from the first compound synthesis in the laboratory until the therapeutic agent, or drug, is brought to market. GriDock is a molecular modeling software developed to identify potentially bioactive compounds (hit-compounds) in order to speed-up the drug discovery process. It is based on the virtual-screening approach in which the activity of a large set of molecules is predicted by multiple molecular docking calculations distributed on a Grid system. More in details, GriDock joints the VEGA flexibility and the AutoDock 4 power to take full advantage of the Grid technology.**
The applications were ported on the Sicilian Grid infrastructure, solving parallelization problem and adjusting compilation time directives. This effort leaded to a ready and run process able to produce results on the Grid.
This paper presents the porting on PI2S2 infrastructure GriDock that aims to perform drug-lead exploration in an efficient and inexpensive manner on computing Grids.

***Index Terms*— docking, Grid, CTV, virtual screening, drug discovery.**

I. INTRODUCTION

The number of proteins with a known three-dimensional structure is increasing rapidly, and structures produced by structural genomics initiatives are beginning to become publicly available. The increase in the number of structural targets is in part due to improvements in techniques for structure determination, such as high throughput X-ray crystallography. With large-scale structure-determination projects driven by genomics consortia, many current target proteins have been selected for their therapeutic potential. Computational methodologies have become a crucial component of many drug discovery programmes, from hit identification to lead optimization and beyond, and approaches such as ligand- or structure-based virtual screening techniques are widely used in many discovery efforts. Indeed, there are now a number of drugs whose development was heavily influenced by or based on structure-based design and screening strategies, such as HIV protease inhibitors [1].

A. Virtual screening: the state of the art.

Ligand- and/or structure-based virtual screenings can be applied to identify new compounds potentially active on specific biological targets. The ligand-based approaches do not require the knowledge of the biological target involved in a disease. These methods can allow robust predictions of biological properties for new chemical entities and rely on the recognition of key features for an optimal biological activity (pharmacophore elements) as well as on the pivotal principle stating that molecules with similar physicochemical properties should also exhibit similar biological activities. These approaches consist of construction of a mathematical model relating chemical properties to

biological effect by means of statistical techniques (QSAR studies). The soundness of the results can be limited by some known pitfalls among which one can consider that (i) different molecules act by different mechanisms or interact with the receptor in different binding modes and (ii) the intrinsic noise associated with both training set, which does not suitably cover the chemical space, and computational aspects involved in the prediction.

The structure-based approaches depend on the availability of structural information of the target, that being either determined experimentally or derived computationally by means of homology modelling techniques. They allow to analyze a wider chemical space, because the structure of the biological target provides reasonable estimations of both expected pose of the ligand into the protein cavity (docking) and of its binding affinity as predicted by docking scores.

B. GriDock: the parallel structure-based virtual screening system.

On this ground, we developed a new structure-based virtual screening software, called *GriDock*, able to run on multi-CPU/core systems and Grid architectures due to its parallel design. In particular, this program, written in highly portable C++ code, can screen a large number of ligands contained in virtual libraries bridging VEGA [2, 3] with AutoDock 4 [4]. The former program manages databases of molecules and prepares the input files required by the latter that performs the molecular docking calculations. GriDock controls the job of both programs, analyzes the results of each docking calculation and stores the data in output files (**Figure 1**).

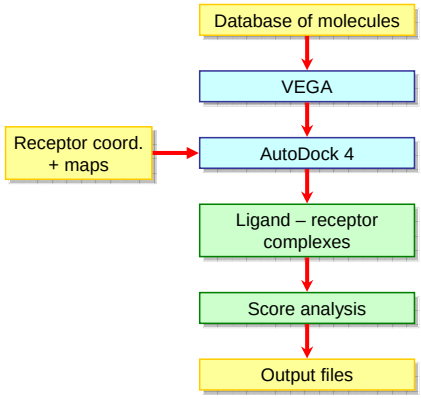


Figure 1: each ligand/molecule contained in the database is extracted and pre-processed by VEGA to prepare the input file for AutoDock 4 that performs the real docking calculation. When the docking is finished, GriDock analyzes the quality of each ligand – receptor complex on the basis of the score. Finally, the best complex is stored in the output files.

More in detail, VEGA converts an input file (in whichever format supported by VEGA) to the PDBQT format as required by AutoDock 4. During this step, some pre-processing operations are performed as shown in **Figure 2**.

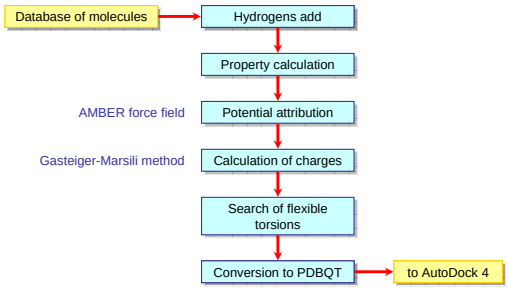


Figure 2: VEGA pre-processes each molecule contained in the database before the final conversion to PDBQT format, adding the hydrogens, calculating some molecular properties required by GriDock, applying the atom potentials (AMBER force field), calculating atom charges (Gasteiger – Marsili method) and searching the flexible torsion angles, used by AutoDock 4 to simulate the ligand flexibility.

All computational steps involved in the virtual screening were designed to be executed in parallel way, allowing to develop multi-

threaded and Grid versions of GriDock. This last version is based on MPICH2 that is a high performance and portable implementation of the message passing interface (MPI) standard. In particular, the MPI master node controls the activity of each worker that screens one molecule at a time included in the database and sends the results to the master node (**Figure 3**). The master node writes three output files: 1) log file in which it is possible to watch the calculation steps; 2) comma separated value file (CSV) containing the list of the ligand-receptor complexes ranked by docking score; 3) zip archive including all 3D structures of the complexes as generated by AutoDock 4.

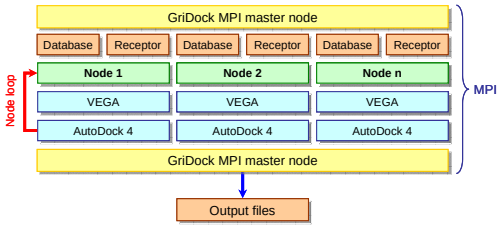


Figure 3: block diagram of GriDock MPI version. The master node watches the activity of each worker that runs a single docking procedure. When the AutoDock 4 calculation is finished, the result is sent to the worker that analyzes it and write the output files.

The input files required to perform a GriDock virtual screening can be listed as follows: (i) the 3D experimental or homology modelled structure of the biological target, (ii) the 3D maps of the binding site generated by AutoGrid 4, included in the AutoDock 4 package, and (iii) one or more databases of 3D structures. The databases should contain a very large number of olecules in order to exhaustively explore the relevant chemical space.

C. Gridock porting on GILDA infrastrucuture

The porting of Gridock on GILDA infrastructure is made by a collection of simple bash scripts. After the usual authentication step, the user is able to submit his simulation using some file properties.

The following picture shows the use case scenario:

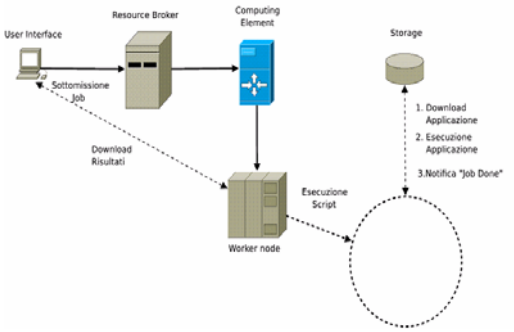


Figure 4: diagram of GriDock Grid version.

The main actions are done by the following files:

1. ***gridock_thread_iceage.jdl***, jdl file describing to the "Resource Broker" all job requirements. Here we specify the execution file.
2. ***gridock.sh***, bash script doing the following steps: download the executable file from the storage (41M), untar downloaded archive, set le all needed environmental variables, run the simulation, make an archivi containg all needed results.

As we can see from the reported bash script, the process is very simple and the approach is usable for MPI version, with a bit of work.

```
#!/bin/bash

echo "Downloading from storage"
/opt/lcg/bin/lcg-cp --vo gilda
lfn:/grid/gilda/users/mastriani/vega.tg
z file:`pwd`/vega.tgz

echo "Untar archive"
tar -xzvf vega.tgz

echo "Setting environment variables"
export VEGADIR=`pwd`
export
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:`pwd`
export PATH=$PATH:`pwd`

echo "Settenv for Gridock is:"
env | grep `pwd`

echo "Changing mode"
chmod +x gridock autodock4 autogrid4
vega
```

```
PWD/./gridock -l 4
Vega/GRIDOCK_DATA/RdRp/GriDock_Test/rdr
p_q2xp15.pdbqt Vega/G
RIDOCK_DATA/RdRp/GriDock_Test/ChemBank.
sdf Vega/Data/Autodock/default.dpf

echo "Tar archive"
tar -czvf out_gridock.tgz gridock*.log
*.csv *.zip
```

D. Citrus Tristeza Virus (CTV): a case of study.

In order to test the features implemented in GriDock, the *Citrus tristeza virus* (CTV) case was considered. CTV is distributed worldwide and is the causal agent of one of the most economically important diseases of citrus. CTV, a member of the genus *Closterovirus* within the family *Closteroviridae*, is phloem limited and is transmitted by aphids in a semipersistent manner. CTV virions are filamentous flexuous particles about 2,000 nm long, with two coat proteins (CP and CPm) covering 95 and 5% of the particle length, respectively. The CTV genome is a single-stranded, positive-sense RNA of 19,226 to 19,296 nucleotides (nt) organized in 12 open reading frames encoding at least 19 proteins [5]. The total economic loss caused by CTV is difficult to evaluate, but a global loss has been estimated to billion of dollars. Around 80 million of *Citrus* trees have been destroyed in the world by CTV infection. Several approches have been used to control damage CTV. The choice of approach is dependent on CTV incidence. In areas where CTV is rare, and natural spread has been limited, the emphasis has been on avoiding the virus. Quarantine and certification programs have been used to prevent introduction and propagation of infected plants into new areas. Eradication programs have also been used where infection levels are low or localized to suppress further natural spread of CTV. In areas where CTV is endemic and risk of natural infection are high, the approach is to control losses to production. Actually doesn't exist a treatment for CTV infections.

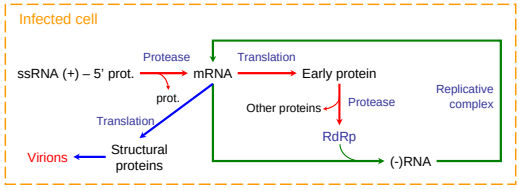


Figure 4: CTV replication cycle in infected cells. Closteroviruses have large complex genomes containing up to 12 ORFs. The first two ORFs overlap and the second is expressed by a + 1 frameshift from the first.The first ORF contains a papain-like protease that processes at two sites. The frameshift between the first two ORFs of CTV and the proteolytic processing give rise to nine polypeptides. The other 10 ORFs are expressed from subgenomic RNAs.

Some researchers identified the RNA-dependent-RNA polymerase (RdRp) as possible therapeutic target because this enzyme plays a pivotal role in the CTV replication (**Figure 4**). For this reason, the aim of this computational research study is the modelling of the viral RdRp and the identification of potential inhibitors of this enzyme through virtual screenings calculations.

Since the structure of CTV RdRp has not yet been resolved, the enzyme was modelled by homology techniques in which the aminoacid sequence downloaded from the SwissProt database [6] (Entry ID: QXP15) has been used as input for the folding prediction as provided by Fugue [7]. This is a on-line service that scans a database of structural profiles, calculates the sequence-structure compatibility scores and produces a list of potential homologue structures and alignments. The best scored model was refined by VEGA ZZ package adding the aminoacidic side chains and the hydrogens. The rough structure was optimized by conjugate gradients minimization (30.000 steps), performed by NAMD software [8]. The refined RdRp model was processed to prepare the AutoDock 4 input, calculating the atomic charges, assigning the AMBER force field, removing the apolar hydrogens and saving the structure in PDBQT format. Finally, the grid maps of the mRNA binding site were generated. Also these operations were performed by VEGA ZZ.

In order to perform the GriDock virtual screening calculations, about one million of molecules included in the Ligand.info Small Molecule Meta-Database [9] were considered. More in detail, they are classified in the following subsets: ChemBank, ChemPDB, KEGG Ligand, Anti-HIV NCI, Drug-likeness NCI, Not annotated NCI, AKos GmbH, Asinex Ltd and TimTec.

Analyzing the screening results, it was possible to highlight that the best scored ligands contain one or more sulphate groups in their structure. Moreover, some of them are know to be potent inhibitors of the HIV reverse transcriptase and are included in the class of the naphtalen polysulfonic acids (**Figure 5**). These results are encouraging, because the HIV reverse transcriptase is RNA polymerase as RdRp, but they need to be confirmed by experimental assays.

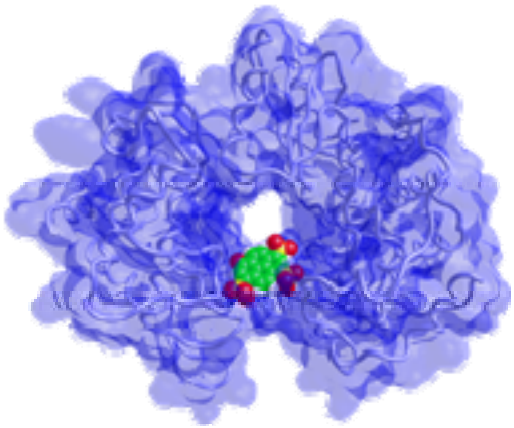


Figure 5: RdRp refined model complexed with 3-hydroxypyren-1,6,8-trisulfonic acid, an anti-HIV drug. The ligand inhibits the RNA access to its binding site due to the strong ionic interactions (-10.83 kcal/mol) between its sulfonic groups and Lys 93, Arg 129 and Lys 369 of the enzyme (not shown in picture).

II. Conclusion

This study presents a new parallel structure-based virtual screening software which can run on both multi-CPU/core and Grid systems.

After the hard startup, the result is a simple method to submit Gridock job to GILDA infrastructure. With a bit of work, it is possible to run MPI-Gridock version under the Grid.

With a view to verify its potentialities, the RdRp enzyme was considered as possible therapeutically target for citruses infected by CTV and its 3D structure was obtained through homology approaches. The screening of more than one million of molecules allowed the identification of some potential RdRp inhibitors that are sulfonic acid derivatives included in the anti-HIV class. The achieved results must be considered in fact preliminary since they have to be confirmed by experimental data. Nonetheless, they can be seen as a good starting point to develop a novel drug class against CTV infections.

Acknowledgments

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it

References

[1] D.B. Kitchen, H. Decornez, J.R. Furr and J. Bajorath, "Docking and scoring in virtual screening for drug discovery: methods and applications", *Nature Reviews Drug Discovery*, **3**, 935-949 (2004).

[2] A. Pedretti, L. Villa, G. Vistoli, "VEGA - An open platform to develop chemo-bio-informatics applications, using Plug-in architecture and script programming", *J. Comput. Aided. Mol. Des.*, **18**, 167-173 (2004).

[3] A. Pedretti, L. Villa, G. Vistoli, "VEGA: a versatile program to convert, handle and visualize molecular structure on windows-based PCs", *J. Mol. Graph.*, **21**, 47-49 (2002).

[4] D.S. Goodsell, G.M. Morris, A.J. Olson, "Automated docking of flexible ligands: applications of AutoDock", *J. Mol. Recognit.*, 9(1), 1-5 (1996).

[5] L. Rubio, M.A. Ayllon, P. Kong, A. Fernadez, M. Polek, J. Guerri, P. Moreno and B.W. Falk
Genetic Variation of *Citrus Tristeza Virus* Isolates from California and Spain: Evidence for Mixed Infections and

Recombination. *Journal of Virology* **75**, 8054–8062 (2001).
[6] ExPASy Proteomics Server, http://www.expasy.org
[7] J. Shi, T.L. Blundell, K. Mizuguchi, "FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties", *J. Mol. Biol.*, **310(1)**, 243-57, (2001).
[8] J.C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale, K. Schulten, "Scalable molecular dynamics with NAMD", *J. Comp. Chem.*, **26**, 1781-1802 (2005).
[9] http://ligand.info

A. Pedretti was born in 1970. He received his Laurea degree in Medicinal Chemistry at University of Milan in 1995. After PhD studies with Prof. Luigi Villa, he became assistant professor in medicinal chemistry at University of Milan in 2001. His interests mainly deal with computer programming applied in computational chemistry, realizing novel software tools for molecular modelling, docking analysis and virtual screening. More in details, he developed VEGA (available at www.vegazz.net), a software package able to build, visualize, optimize and analyze 3D structures. VEGA can also compute several molecular fields analyzing their dynamic profiles during the simulation time. He is author of more than 30 scientific papers published in international refereed journals and 5 monographs in books.

A. Lombardo was born in 1971. He is a molecular biologist with a P.h.D in "Phytosanitary technologies" on biotechnological applications in plant pathology at the University of Catania.
He is member of the steering committee of the italian technological platform "IT-Plants for the future", member of the scientific committee, challenge conversion, of the italian technological platform "Biofuels Italy".
Since 2005 he is responsible of the laboratory of biotechnology, genomic analysis and varietal correspondence (Science and Technology Park of Sicily) and since 2008 responsible of the laboratory of phytopathology (Science and Technology Park of Sicily). He was involved in several Interregional, national and regional projects. He is author of about 25 scientific papers published in international refereed journals and presented in national and international congress and 1 patent.
His main research areas are the development of expression vectors for plants, yeasts and bacteria, bioinformatic applied on plant viruses, variety correspondence, molecular characterization of plants, yeasts and viruses.

G. Vistoli was born in 1968. He received his Laurea degree in medicinal chemistry at University of Milan in 1994. During his PhD studies with Prof. L. Villa, he spent a period in Lausanne under the supervision of Prof. B. Testa with which he has fruitfully collaborated since 1996. In 1999, he became assistant professor in medicinal chemistry at University of Milan. His recent research interests focus on (i) the development of property space concept to explore the dynamic profile of molecular fields deriving fertile descriptors for dynamic 4D-QSAR analyses; (ii) modeling of transmembrane proteins (with particular attention to GPCRs) using innovative approaches able to account for the local peculiarities (fragmental approach); (iii) in silico analyses of functional and structural effects of covalent adduct on physiologically relevant proteins; (iv) rational design of RCS scavengers combining *ab initio* and molecular dynamics methods.

E. Mastriani During my studies, I have carried out several researches in the Computer Security and Artificial Intelligence fields, obtaining relevant published results. I have worked in Operating System Security, producing "Piranha Audit". I am currently site administrator of the Grid-node of Department of Mathematics and Computer Science (University of Catania). I also offer support for the process of adapting of scientific and industrial applications so that it can run over the Grid infrastructure. I am specializing on the use of the Grid technology and my latest courses are: 5th International Summer School on Grid Computing, 3rd DEISA Training Session .I am also studying models, mathematical methods and computer simulation of cancer growth and therapy. I am a member of: PI2S2 project, CI&IG research group and IMMUNOGRID EC project.

A Variance Reduction Scheme for Monte Carlo Semiconductor Simulations using the PI2S2 Grid

Orazio Muscato[1,2] and Vincenza Di Stefano[1]
[1]*Dipartimento di Matematica e Informatica, viale A. Doria 6 , 95125 Catania, Italy*
[2]*Consorzio COMETA, Via S.Sofia 78, I-95123 Catania, Italy*
{*muscato,vdistefano*}*@dmi.unict.it*

***Abstract*—The Multicomb variance reduction technique has been introduced in the Direct Monte Carlo Simulation for submicrometric semiconductors. We have implemented the method in bulk silicon and demonstrated its effectiveness in the study of hot electron effects. Our simulations show that the variance of hot electrons is reduced with a small computational cost.**

***Index Terms*—Monte Carlo, semiconductors, statistical enhancement, variance reduction.**

I. Introduction

THE continued miniaturization of integrated circuits and the current trend toward nanoscale electronics have led to tremendous integration level, with hundreds of million of transistor assembled on a chip area no larger then few square centimeters. Modern device technologies operate at the length scales close to the electron mean free path (approximately 10 nm in bulk silicon at room temperature). If the active region of the device is order 10-100 nm, ballistic or quasi-ballistic conditions dominate the electron transport, which is far from thermal equilibrium. Moreover large electric fields, which rapidly changes over small length scales, give rise to nonlocal and hot-carrier effects which begin to dominate device performances. The semiclassical Boltzmann transport equation (BTE) [1], coupled with the Poisson equation, provides a general theoretical framework for modeling these transport phenomena. To solve the BTE is a difficult task because it is an integro-differential equation with six dimensions in the phase space and one in time. A statistical solution of the BTE can be obtained with the Direct Simulation Monte Carlo (DSMC) in which the distribution function is replaced with a representative set of particles. The success of the DSMC is due to its flexibility and extensibility. In fact, by using the same algorithm, several materials can be simulated (by considering the appropriate band structure) and scattering mechanisms included (or excluded). Moreover, the DSMC shows itself to be a very useful tool, since it permits particular physical simulations unattainable in experiments, or even investigations of nonexistent materials in order to emphasize special features of the phenomenon under study. However, in exchange for these advantages, significantly longer computer execution times are required in order to minimize the noisy data. In order to produce bona fide simulations, the statistical enhancement (or variance reduction) has been introduced in DSMC partly to compensate for a smaller number of simulated particles as compared to the larger number of actual particles in a device. In this paper we have studied the *Multicomb* variance reduction method. In particular in section II we introduce the Direct Simulation Monte Carlo. In section III the *Multicomb* algorithm is formulated and simulation results are shown for bulk silicon in section IV. Finally conclusions

are drawn in section V.

II. The Direct Simulation Monte Carlo

The Monte Carlo method for evolving a solution of the Boltzmann transport equation consists in recreating the history evolution of electrons in time and space inside the crystal, subject to the action of external and self-consistent electric field and of the given scattering mechanisms [2], [3]. The simulation starts with one or more electrons in given initial conditions with momentum $\hbar\mathsf{k}$ and position x. During the *free flight* (i.e. the time between two collisions) the external forces are made to act according to the Newton's equations of motion in the crystal :

$$\frac{d\mathsf{x}}{dt} = \frac{1}{\hbar}\nabla_{\mathsf{k}}\varepsilon(\mathsf{k}) \quad (1)$$

$$\hbar\frac{d\mathsf{k}}{dt} = -q\mathsf{E}(t,\mathsf{x}) \quad (2)$$

where $\varepsilon(\mathsf{k})$ is the kinetic energy of the considered crystal conduction band structure measured from the band minimum, q the absolute value of the electron charge, and $\hbar$ the Planck constant divided by 2π. The electric field $\mathsf{E}(t,\mathsf{x})$ satisfies the Poisson equation

$$\Delta(\epsilon\phi) = q\Big[n(t,\mathsf{x}) - p(t,\mathsf{x}) - N_D + N_A\Big], \quad (3)$$

$$\mathsf{E} = -\nabla_{\mathsf{x}}\phi,$$

where $\phi(t,\mathsf{x})$ is the electric potential, N_D and N_A are respectively the donor and acceptor densities (which are positive functions), ϵ the permittivity, n and p are respectively the electron and hole densities. In the neighborhood of the band minimum a good dispersion relation is given by the *quasi parabolic* approximation:

$$\varepsilon(\mathsf{k})[1+\alpha\varepsilon(\mathsf{k})] = \frac{\hbar^2\mathsf{k}^2}{2m^\star}, \quad \mathsf{k}\in\Omega, \quad (4)$$

where $m^\star$ is the effective electron mass (which is 0.32 m_e in silicon), α is the nonparabolicity factor, and k belongs to the domain Ω, called first Brillouin zone, which is a characteristic of each material. In silicon this zone is formed by six equivalent ellipsoidal valleys along the axis of the frame of reference at about 0.85 (in the units $\frac{2\pi}{a}$ where a is the lattice constant) from the zone center.

The electron group velocity $\mathsf{v}\equiv(v^1,v^2,v^3)$ is given by

$$\mathsf{v}(\mathsf{k}) = \frac{1}{\hbar}\nabla_{\mathsf{k}}\varepsilon = \frac{\hbar\mathsf{k}}{m^\star[1+2\alpha\varepsilon(\mathsf{k})]} \quad .$$

The equations (1),(2),(3) are solved with a stable numerical scheme by using an appropriate time step Δt [4]. Then a scattering mechanism is chosen randomly as responsible for the end of the free flight, according to the relative probabilities of all possible scattering mechanisms. From the differential cross section of this mechanism a new k state after scattering is randomly chosen as initial state of the new free flight. After the collision the electron can remain in the same valley (intravalley scattering) or be drawn in another equivalent valley (intervalley scattering).

The electrons can scatter by themselves, with the impurities and the lattice. The electron-electron interaction is considered in the framework of the mean field approximation through the Poisson equation. This is reasonable since we consider the case of low doping and therefore we can neglect the short range collisions between electrons. The scattering between electron and the lattice is quantized, and is modeled as a scattering with a particle called *phonon* (from a greek word which means sound). In our code we have considered the scattering with the acoustic (in the elastic approximation) and non polar optical phonons, which are, at room temperature, the main scattering mechanisms in silicon [2].

III. The Algorithm

The most important classes of variance reduction techniques are the *population control* and the *event bias* techniques [5], [6], [7]. In the former, the particles distribution and the weights are adjusted to populate interesting regions of the phase space whereas, in the latter, the scattering and the particle weights

are modified. Each technique has its strengths and weakness. We have chosen the population control technique because of its simplicity and user-friendliness in the implementation. In this class, we have used the so called *Multicomb* algorithm which has been introduced, in the beginning, in the field of neutral particle transport. Let us suppose to have N initial particles with momentum $\hbar \mathsf{k}_i$, energy ε_i and weight w_i. At time zero we choose

$$w_i = \frac{1}{N} \quad .$$

Then we divide the energy space in K bins (or stat-boxes), whose extension is

$$\Delta\varepsilon = \frac{\varepsilon_M}{K}$$

where ε_M is the estimated max energy reached by all particles during the run. So, the energy space is partitioned into $K+1$ bins, i.e.

$$[0, \Delta\varepsilon[, ..., [(K-1)\Delta\varepsilon, K\Delta\varepsilon[, [K\Delta\varepsilon, +\infty[.$$

For each j-th bin we can define :

- the number of particles N_j in the bin , such that
$$N = \sum_{j=1,K+1} N_j$$
- the total weight, as the sum of the weights of the particles which are in the j-th bin
$$W_j = \sum_i w_i \quad , j = 1, .., K+1 \quad .$$

Let's suppose we have reached the stationary regime. Then we decide to run the enhancement algorithm each Δt_{enh}, which we shall call enhancement time step. Let be N_b the number of non-empty bins. Then we define

$$M = \text{int}\left[\frac{N}{Nb}\right] \quad , \quad rest = N - M \times Nb$$

where M is the target particle number in each bin (of course, by definition $M \leq N$), and $rest$ is the division rest. In order to have constant particle number N during the simulation, we make this assumption :

$$M_1 = M + rest, \quad M_2 = M, .., M_{Nb} = M$$

where M_1 is the target particle number in the first bin and so on.

The enhancement algorithm applied to each non-empty bin is called *simple comb*. Let us consider the j-th non-empty bin: this is the *simple comb* algorithm

1) we construct a comb of length W_j with M equally spaced teeth. The position of the m-th tooth is given by
$$t_m = (\xi + m - 1)\frac{W_j}{M_j}$$
where ξ is an uniform random number in [0, 1).
2) Place the weights w_i consecutively on a line segment, obtaining N_j bins
$$[0, w_1],]w_1, w_2],]w_2, w_3], ...,]w_{N_j-1}, w_{N_j}]$$
3) Now we "comb" this line segment with the previous one, obtained in 1).
do $m = 1, .., M_j$
if $t_m \in]w_{i-1}, w_i]$, for some i=1, ..,N_j
a copy of the i-th particle from the old distribution is added to the new distribution, with assigned weight
$$w_i' = \frac{W_j}{M_j}$$
else no copy.
enddo

The algorithm maintains constant, in the j-th bin, the total weight W_j and particle number N_j. It is possible to prove that [5]:

- the *simple comb* preserves on average the individual weights of the pre-combed particles;
- the distribution of particles with identical weights produced by the comb gives the smaller variance than any distribution with unequal weights.

The application of different *simple comb* to all non-empty bins is called *Multicomb* method. Consequently, during the simulation, the *Multicomb* maintains constant (by default) the total particles number N, and the sum of the overall

weights

$$W = \sum_{j=1}^{K+1} W_j \quad .$$

IV. RESULTS

We have tested the *Multicomb* algorithm in a bulk silicon semiconductor, doped to a density of 10^{15}cm^{-3}, using a total target ensemble size of 11000 computational electrons, in which a constant electric field has been frozen.

To determine the electron energy distribution (EED) function, we have discretized the whole energy space in a system of concentric shells with increasing radius

$$\rho_i = i\,h, \quad i = 1, ..., K, \quad h = \frac{\varepsilon_M}{K}$$

(in our case ε_M= 2 eV and K = 200) and by counting the number of particles which are in the corresponding shells, i.e.

$$\begin{aligned} f_1 &= \{\#\,\text{particles} : \varepsilon < \rho_1\} \\ f_i &= \{\#\,\text{particles} : \rho_{i-1} \le \varepsilon < \rho_i\} \\ f_{N_\varepsilon+1} &= \{\#\,\text{particles} : \varepsilon \ge \varepsilon_M\} \quad . \end{aligned}$$

The confidence interval for each point of the EED is obtained by means of the Central Limit Theorem. Let us suppose to make N_r independent runs (repetitions). Then we obtain f_{ij} quantities (i=1,...,K , j=1,...,N_r).

For simplicity we shall call $\xi_j = f_{ij}$ omitting the index i. The confidence interval is evaluated as

$$\bar{\xi} \pm S_\xi$$

with

$$\bar{\xi} = \frac{1}{N_r} \sum_{j=1}^{N_r} \xi_j$$

$$S_\xi = 3 \sqrt{\frac{1}{N_r} \left(\frac{1}{N_r} \sum_{j=1}^{N_r} \xi_j^2 - \left[\frac{1}{N_r} \sum_{j=1}^{N_r} \xi_j \right]^2 \right)}$$

where the factor 3 corresponds to a 99.7 % confidence level. In our case, each repetition had a duration of 20 ps, but data were discarded during the 5 ps initial transient.

In the figures 1,2 we plot the EED functions with the confidence interval, obtained with the *Multicomb* algorithm and without (*Unenhanced*), where an electric field of 48.000 and 120.000 V/cm have been frozen. The accuracy of the Multicomb method is verified by the fact that its errors fall within the error of the unenhanced method (note that the error bars appear asymmetrical about the mean because the vertical axis has a logarithmic scale).

In the figure 1 the maximum energy reached during the unenhanced simulation is 0.44 eV, whereas that with the Multicomb is 0.52 eV, obtaining an enhancement of the tail.

The Relative Error

$$\text{RE} = \frac{S_\xi}{\bar{\xi}}$$

is plotted in the figures 3,4 showing that the Multicomb method gives a lower error at high energies (i.e. $\ge$ 0.4 eV for an electric field of 48.000 V/cm, $\ge$ 1.2 eV for an electric field of 120.000 V/cm). Although the relative error provides useful information about the performance of the methods, it does not take into account the CPU time used. In general, the variance reduced methods pay the price of a greater CPU time consumption, with respect to the unehnanced ones because more control instruction must be added in the code. For example, for an electric field of 48.000 V/cm, the CPU time for Multicomb has been 12100 sec and that for the unenhanced 5200 sec. A useful parameter for comparing the performance of the two methods is the *Figure of Merit*, that takes into account the relative error as well as the CPU time. We define the Figure of Merit (FoM), for the EED, as

$$\text{FoM} = \frac{1}{(\text{RE})^2\, T_{CPU}}$$

where T_{CPU} is the total CPU time. Since the Relative Error is proportional to $1/\sqrt{Nr}$ and the time T_{CPU}, that takes to run Nr runs, is proportional to Nr, then the FoM is independent of Nr. Because the FoM is inversely

proportional to the total CPU time, a method which approaches a given level of error faster will have a higher figure of merit.

In the figures 5, 6 we plot the FoM. The general trend for both methods is that the computational time required for the accurate estimation of the EED exponentially increases with the energy.

At higher energies the Multicomb method is better then the unenhanced one.

Another way to compare the efficiency is to run the unenhanced code for a longer simulation time T_{sim} (i.e. with a longer CPU time), to have the same maximum energy (or the same EED tail) of the *Multicomb*. We have the following table, obtained with an electric field of 48.000 V/cm

CPU in PI2S2 GRID environment

Method	Enh	Enh	Multi
T_{sim} (ps)	20	140	20
Max ene (eV)	0.44	0.52	0.52
T_{CPU} (sec)	5200	36400	12100

where Enh and Multi indicate respectively the Enhanced and Multicomb methods, Max ene is the maximum energy reached during the run, and T_{sim} is the total simulation time.

In order to reach the same max energy, the unenhanced algorithm must run for a longer simulation time (T_{sim}= 140 ps), consuming 36400 sec of CPU, which means a factor 3 with respect to the Multicomb algorithm, whose CPU time is 12100 sec. Similar results have been obtained with a higher electric field.

V. Conclusion

Monte Carlo semiconductor simulations need some form of variance reduction to minimize the noise in the data. We have implemented the *Multicomb* algorithm, in the case of bulk silicon. The use of this algorithm can enhance the population of high energy electrons and give better information about the high energy distribution. In order to have the same accuracy in the tail of the EED, the CPU consumed by the unenhanced algorithm must be 3 times that of the *Multicomb* one.

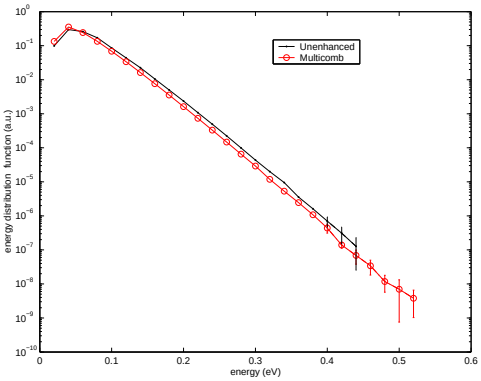


Fig. 1. The energy distribution with error bar, for the unenhanced and multicomb methods, averaged over 100 simulations of 20 ps and using 11000 electrons, obtained with an electric field of 48000 V/cm.

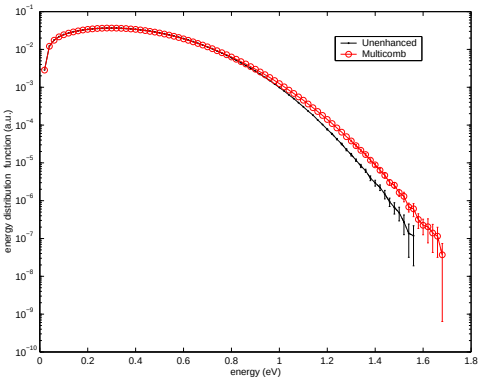


Fig. 2. The energy distribution with error bar, for the unenhanced and multicomb methods, averaged over 100 simulations of 20 ps and using 11000 electrons, obtained with an electric field of 120000 V/cm.

Acknowledgment

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Programma Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it. The authors acknowledge the Pi2S2 system managers for their patience and expertise in solving problems, and in particular those of the UNICT-DMI, UNICT-

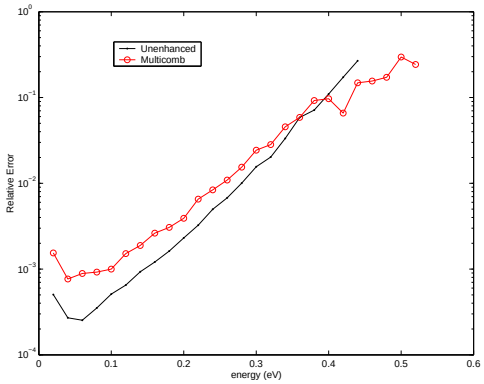


Fig. 3. Relative error for the energy distribution versus energy for the unenhanced and multicomb methods, averaged over 100 simulations of 20 ps and using 11000 electrons, obtained with an electric field of 48000 V/cm

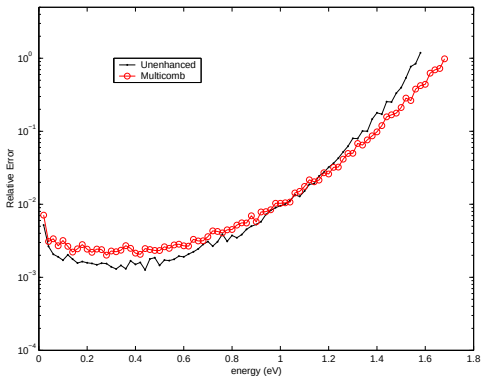


Fig. 4. Relative error for the energy distribution versus energy for the unenhanced and multicomb methods, averaged over 100 simulations of 20 ps and using 11000 electrons, obtained with an electric field of 120000 V/cm

DIIT, and INFN-CT sites.

REFERENCES

[1] P. A. Markowich, C. A. Ringhofer, and C. Schmeiser, *Semiconductor equations*. Vienna: Springer-Verlag, 1990.

[2] C. Jacoboni and L. Reggiani, "The Monte Carlo method for the solution of charge transport in semiconductors with applications to covalent materials," *Rev. Modern Phys.*, vol. 55, no. 3, pp. 645–705, 1983.

[3] K. Tomizawa, *Numerical Simulation of Submicron Semiconductor Devices*. Boston: Arthech House, 1993.

[4] O. Muscato and W. Wagner, "Time step truncation in direct simulation Monte Carlo for semiconductors," *COMPEL*, vol. 24, no. 4, pp. 1351–1366, 2005.

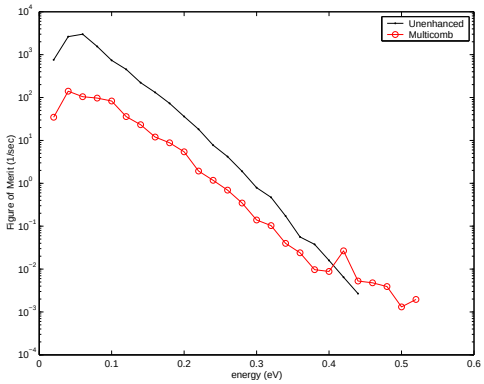


Fig. 5. Figure of merit versus the energy for the unenhanced and multicomb methods, using 11000 electrons, obtained with an electric field of 48000 V/cm

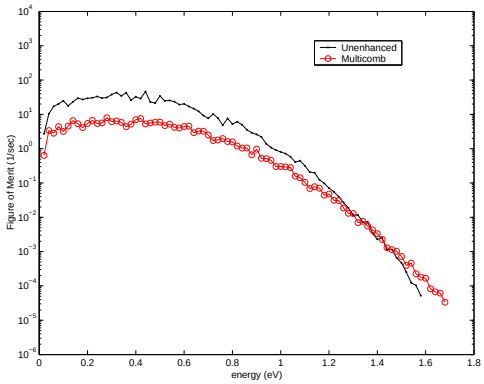


Fig. 6. Figure of merit versus the energy for the unenhanced and multicomb methods, using 11000 electrons, obtained with an electric field of 120000 V/cm

[5] M. G. Gray, T. E. Booth, T. J. T. Kwan, and C. M. Snell, "A Multicomb variance reduction scheme for Monte Carlo semiconductor simulators," *IEEE Trans. Elec. Dev.*, vol. 45, no. 4, pp. 918–924, 1998.

[6] C. J. Wordelman, T. E. Booth, and C. M. Snell, "Comparison of statistical enhancement methods for Monte Carlo semiconductor simulations," vol. 17, no. 12, pp. 1230–1235, 1998.

[7] M. Nedjalkov, H. Kosina, and S. Selberherr, "The stationary Monte Carlo method for device simulation.II. Event biasing and variance estimation," vol. 93, no. 6, pp. 3564–3571, 2003.

Orazio Muscato Orazio Muscato is Associate Professor of Mathematical Physics at Catania University. His research interests include mathematical models for semiconductors, wave propagation.

Vincenza Di Stefano Vincenza Di Stefano is Post-doc at Catania University. Her research interests include mathematical models for semiconductors.

Materials for Hydrogen Storage: an ab-initio Molecular Dynamics Study in the ENEA-GRID Environment

[1]M. Celino and [1]S. Giusepponi
[1]ENEA, Dipartimento Tecnologie Fisiche e Nuovi Materiali
C. R. Casaccia, Via Anguillarese 301, 00123, S. Maria di Galeria (Roma)

Abstract—**The development of a viable hydrogen storage system is becoming increasingly important for promoting the "hydrogen economy". Several materials are good candidate for storing hydrogen, but, among them, magnesium hydride MgH_2 is one of the most promising candidates as hydrogen storage media in automotive industry, due to its very high capacity, light weight and low cost of production. However, further research is needed since Mg has a high operation temperature and slow absorption kinetics that prevent for the moment the use in practical applications. Experimentally, great efforts have been made to improve the H_2 absorption and desorption kinetics by mechanically milling MgH_2-Mg with various additives. However the atomic level mechanism by which hydrogen flows in and out of the hydride along the MgH_2-Mg interfaces still remain unclear. In order to have a better understanding of this mechanism, extensive *ab-initio* molecular dynamics simulations have been performed to gain detailed information of the atomic level dynamics during desorption. Electronic structure calculations are used to determine the equilibrium properties and the behavior of the surfaces in terms of total energy considerations and varying the external temperature. Furthermore, metallic additives are added to the MgH_2 system in order to characterize the atomic origin of the improved hydrogen diffusion at the interface.**

I. Introduction

THE development of a viable hydrogen storage system is becoming increasingly important for promoting the "hydrogen economy". Several materials are good candidate for storing hydrogen, but, among them, magnesium hydride MgH_2 is one of the most promising candidates as hydrogen storage media in automotive industry, due to its very high capacity (7.7 wt%), light weight and low cost of production. However, further research is needed since Mg has a high operation temperature and slow absorption kinetics that prevent for the moment the use in practical applications. Experimentally, great efforts have been made to improve the H_2 absorption and desorption kinetics by mechanically milling MgH_2 with various additives [1]–[3]. However the atomic level mechanism by which hydrogen flows in and out of the hydride along the MgH_2-Mg interfaces still remains unclear [4]. In order to have a better understanding of this mechanism, extensive *ab-initio* Molecular Dynamics (*MD*) simulations have been performed to gain detailed information of the atomic level dynamics during desorption. Electronic structure calculations are used to determine the equilibrium properties and the behavior of the surfaces in terms of total energy considerations. Furthermore, the interface is studied at several values of the temperature, thereby characterizing the hydrogen atomic displacement. The hydrogen desorption happens at the same temperatures measured experimentally. To gain more insights on the role played by catalytic metals in MgH_2 a Mg atom is replaced by a Fe atom. Accordingly with experiments, the temperature of H desorption is lowered significantly. Moreover the mechanism for which the metallic atoms

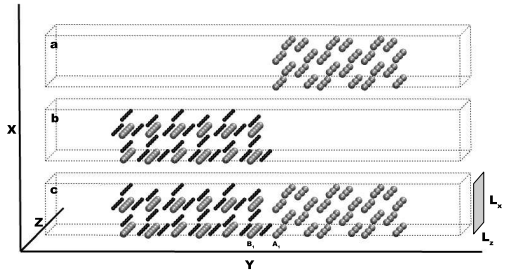


Fig. 1: Simulation cell for first-principle molecular dynamics simulations: a) *Mg* bulk with the internal surface on the left; b) MgH_2 with the internal surface on the right; c) Starting configuration of the MgH_2-*Mg* interface.

act to free hydrogen atoms is characterized and measured.

II. CPMD CODE

We employed for all the calculations the *CPMD* (Car-Parrinello Molecular Dynamics) code [5], [6]. *CPMD* is an *ab-initio* electronic structure and MD program using a plane wave/pseudopotential implementation of density functional theory. The CPMD code is based on the original computer code written by Car and Parrinello [5]. During the last years the code has grown and now it has about 200,000 lines of code. CPMD is currently used by thousands of users in more than 50 countries.

Ab-initio MD is the combination of first-principle electronic structure methods with MD based on Newton's equation of motion. Electronic approaches from first-principle overcome the shortcomings of parametric interatomic potentials allowing to adjust to new chemical atomic configurations that may be encountered during a simulation. This assures the reliability of the dinamical evolution of the atomic system even when chemical reactions or structural phase transitions occur.

In their seminal paper Car and Parrinello [5] introduced a new method to efficiently propagate the electonic wave-function togheter with the atomic cores. The theoretical approach is based on the solution of the Kohn-Sham equation of the Density Functional Theory. This method has been proved to be valuable in many fields. We can recall solid-state physics, surface science, intefaces, glasses and amorphous systems, liquids and solutions, catalysis and other chemical reactions, as well as problems from biophysics and biochemistry. Due to the ab-initio nature of the code, many parallel implementations have been developed to push the applications from the originally few atoms systems to several hundreds atoms projects.

The Car and Parrinello method is based on the extended Langrangian

$$\begin{aligned}\mathcal{L}_{cl} &= \tfrac{1}{2}\sum_I M_I \dot{\mathbf{R}}_I^2 + \tfrac{1}{2}\sum_i \mu_i \langle\dot{\psi}_i|\dot{\psi}_i\rangle \\ &\quad -E_{KS}\left[\{\psi_i\},\mathbf{R}_I\right] \\ &\quad +\sum_{ij}\Lambda_{ij}(\langle\dot{\psi}_i|\dot{\psi}_i\rangle - \delta_{ij})\end{aligned} \tag{1}$$

where M_I and R_I are the mass and the position of nuclei I, μ is a fictitious electron mass, ψ_i are the Kohn-Sham orbitals, and E_{KS} is the Kohn-Sham energy. The last term is a holomic constraint to ensure orthogonality of the orbitals. From the Lagrangian, the equations of motion can be derived

$$M_I\ddot{\mathbf{R}}_\mathbf{I}(t) = -\frac{\partial}{\partial\mathbf{R}_I}E_{KS}\left[\{\psi_i\},\mathbf{R}_I\right] \tag{2}$$

$$\mu_I\ddot{\psi}_i(t) = -\frac{\delta}{\delta\psi_i^*}E_{KS}\left[\{\psi_i\},\mathbf{R}_I\right] + \sum_j \Lambda_{ij}\psi_j \tag{3}$$

The equation of motion are integrated using a velocity Verlet scheme. This makes it necessary to compute the forces on the ions $\partial/\partial\mathbf{R}_I(E_{KS})$ at each time step during the simulation and the orbitals $\delta/\delta\psi_i^*(E_{KS})$, as well as to determine the Lagrange moltipliers Λ_{ij}.

The total energy of the system is measured

by the Kohn-Sham energy

$$\begin{aligned} E_{KS} &= \sum_i^{occ} f_i \int dr\, \psi_i^*(r) - \tfrac{1}{2}\nabla^2\psi_i(r) \\ &+ \int dr\, V_{ext}(r)\rho(r) + \tfrac{1}{2}\int\int dr\, dr' \tfrac{\rho(r)\rho(r')}{|r-r'|} \\ &+ E_{XC} + E_I[\{R_I\}] \end{aligned} \tag{4}$$

where $V_{ext}(r)$ is the ionic potential,

$$\rho(r) = \sum_i f_i |\psi_i(r)|^2 \tag{5}$$

the electronic charge density, f_i the occupation numbers, E_{XC} the exchange-correlation energy, and E_I the electrostatic ion-ion interaction. Periodic boundary conditions allow the expansion of the wave function ψ_i in planes waves.

The Kohn-Sham orbitals can be written as

$$\psi_i(r) = \frac{1}{\sqrt{\Omega}} \sum_G c_i(G) \exp(iGr) \tag{6}$$

where Ω is the volume of the simulation cell, G the reciprocal lattice vectors, and c_i the Fourier coefficients of the orbital i. This sum is reduced to a finite set by truncating the sum over G to include only those plane waves with a specified kinetic energy. Given these equations, the forces can be computed on the ions and on the coefficients of the electronic orbitals. Thus they can evolve in time on the same foot.

Fast Fourier transforms (FFT) are an essential part of all plane-wave calculations. In fact, it is the linear scaling of FFT's that makes large-scale density-functional theory calculations with plane-wave basis sets possible. FFT's are used to transform the charge density and local potential between real space grid and reciprocal space. The number of these transforms is fixed and does not depend on the system size. On the other hand the transforms of wave functions from reciprocal space to real space and back for the force calculations has to be done for each state and therefore scales with the number of electrons. The overall scaling is NMlogM and is predominant for small systems. Only for large systems (number of atoms greater than 1000) become predominant. Optimized 3D FFT kernels are implemented by using 1D FFT routines to overcome the intrinsic sparsity of the wave functions arrays.

III. Parallelization of CPMD

Parallelization of CPMD code is done on different levels. The central parallelization is based on a distributed memory coarse grained algorithm, which is a compromise among load balancing, memory distribution and parallel efficiency. This scheme achieves good performance on computers with up to 200 CPUs, depending on system size and communication speed. To overcome to this limited scalability on top of the basic scheme a fine-grain, shared memory parallelization is implemented. The two parallelization methods are independent and can be mixed. This allows us to achieve good performance on distributed computers with shared-memory nodes and several thousands of CPUs.

Distributed memory parallelization is driven by the distribution of wave function coefficients for all states to all CPUs. Real space grids are also distributed, whereas all matrices that do not include a plane-wave index are replicated. In this manner all loops over plan waves, especially the ones having N^2M scaling (overlap matrices, orbital rotations), are parallel.

A parallel 3D FFT is also required. The 3D FFT implements: 1) scatter of data; 2) 1D transforms along direction x; 3) 1D transforms along direction y; 4) 1D transforms along direction z. For a general data distribution in both spaces each of the above steps would include communication between all processors. The data distribution in CPMD tries to minimize the number of communication steps while maintaining optimum load balancing in both spaces. In this way only a single data communication step after the first 1D transform. In addition, we can make us of the sparsity of the wave function representation still present after the first transform and only communicate nonzero elements. The vorious load balancing requirements are interrelated and an heuristic algorithm is used to improve performances.

On the contrary shared memory parallelization (OpenMP compiler directives and multi-threaded libraries are used) is used to parallelize internal loops. Compiler directives are used to ensure parallelization of all longer loops (those

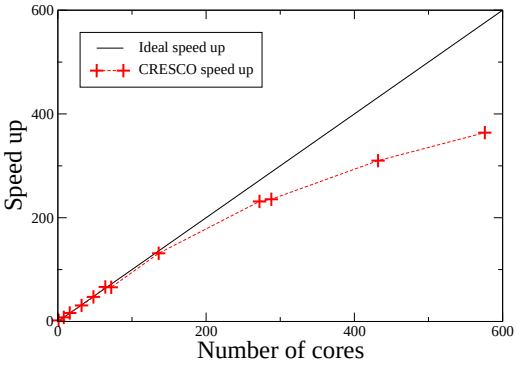


Fig. 2: Speed up of *CPMD* code *versus* number of cores in *HPC CRESCO*

that depend on the number of plane waves or the number of grids points in real space), and to avoid parallelization on the shorter ones. This type of parallelization is independent of the MPI parallelization and can be used alone or combined with the distribution memory approach. Tests on various shared memory computers have shown that efficient parallelization up to 16 processors can be achieved. It is not surprising that loop-level parallelism is very effective in CPMD. This approach is interesting for the following two reasons. First, the shared memory parallelization is also effective in the serial part of the distributed memory scheme, e.g. the rotation matrix and overlap matrices. Second, for a given number of processors, the number of tasks involved in the distributed memory parallelization can easily be decreased by one order of magnitude, reducing the impact of the latency in the all-to-all communications and obtaining good scaling behavior for up to thousands of processors.

IV. Computational details

Goedecker-Teter-Hutter pseudopotentials for magnesium and hydrogen, together with Padé approximant *LDA* exchange-correlation potentials are used [7]. The electronic wave functions are expanded in a plane-wave basis set with a kinetic energy cut-off equal to 80 *Ry*. The latter value was optimized by preliminary calculations both on simple molecules (Mg_2, MgH and H_2), and on the crystalline structures of metallic Mg and magnesium hydride. All the calculations are performed in the supercell approximation, in view of the large-scale molecular dynamics simulation of the interface, with periodic boundary conditions meant to mimic an infinitely extended system [8]. The crystal structure of Mg is the hexagonal close-packed and the crystal MgH_2 is considered in the β-MgH_2 atomic structure, TiO_2-rutile type, that is observed at the atmospheric pressure and low temperatures [9].

We test the *CPMD* code by using the system showed in Fig. 1c representing a MgH_2-Mg interface constituted by 132 atoms of magnesium and 120 atoms of hydrogen. We use a *CPMD* (version 3.11.1) code compiled with Intel Fortran Compiler (v10.1), *MKL* library and *MPI* parallelization on the high performance computer *CRESCO* [11]. In the Fig. 2 the speed-up calculated from the total time required by the code to perform 300 MD time steps using increasing number of cores (from 8 up to 576) is reported. The benchmark results demonstrate the high performance of the CPMD parallelization on *CRESCO* architecture and the good scalability up to hundreds of cores.

V. Model interface

A Supercell approach is used to simulate the atomic system with the interface [10]. As shown in Fig. 1, the MgH_2-Mg interface is built by putting nearby two free surfaces obtained cutting both the Mg and MgH_2 crystals. The main constraint in building the interface is in the selection of two commensurate surfaces (one for Mg and one for MgH_2) fulfilling a simulation cell with periodic boundary conditions (see Fig. 1). Such a constraint narrows the possibilities of finding two suitable free surfaces for a proper interface. The interface considered in this work is the one composed by facing the (010) of Mg with the (110) of MgH_2.

The Mg part is sketched in Fig. 1a and the MgH_2 one in Fig. 1b: these systems can interact through their internal free surfaces forming an

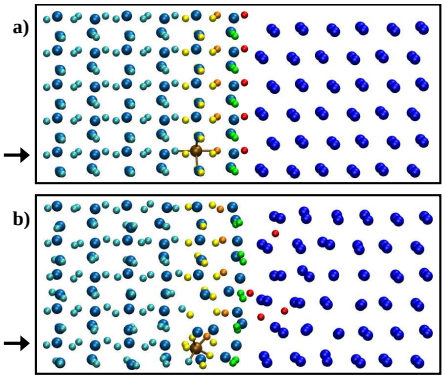


Fig. 3: Simulation cell for first-principle molecular dynamics simulations of the catalytic effect of a Fe atom near a MgH_2-Mg interface. Small atoms are hydrogen, the others are Mg except the atom with bonded H that is the Fe atom. The arrows indicate the horizontal line where the Fe is inserted. a) Starting configuration. b) Snapshot at T= 400 K.

interface as shown in Fig. 1c (132 *Mg* atoms and 120 *H* atoms). Each free surface, selected according to periodic boundary conditions, is relaxed without taking into account any possible reconstruction upon changing the temperature. The total length of the system is L_y= 50.30 *Å*, while in the x and z direction the system has L_x= 6.21 *Å* and L_z= 15.09 *Å*, respectively. On both sides of the system, a void region of length $2 \times L_x$ is considered to suppress the interaction, due to periodic boundary conditions, between the external surfaces of *Mg* and MgH_2 that are kept fixed. Zero-temperature total energy calculations are used to evaluate interface stability and *MD* simulations, at constant volume and constant temperature, are performed to characterize the hydrogen diffusion [10]. To understand the atomic level dynamics of hydrogen diffusion at the interface, *MD* simulations starting from room temperature (T= 300 K) and up to T= 900 K are performed. In this range of temperature, a phase transition from MgH_2 to $Mg + H_2$ is experimentally observed, accompanied by the desorption of hydrogen. Molecular dynamics simulations, performed from room temperature to 900 *K*, reveal an increased mobility of hydrogen atoms near the interface. Furthermore, diffusion of hydrogen atoms starts at T= 700 K, in agreement with experimental observations, confirming the accuratness of the *ab-initio* approach. The mobility of interface hydrogen is much higher than for bulk hydrogen atoms. This difference increases at higher temperatures and no hydrogen atoms are observed to diffuse in the *Mg* bulk.

Finally, a Mg atom in the MgH_2 part is replaced by a Fe atom to simulate the catalytic process seen in the experiments. A Fe atom is inserted in the second row from the interface (Fig. 3). MD simulation at T=400 K clearly shows that the hydrogen diffusion already starts. This is in very good agreement with experiments shown above. Moreover, Fig. 3 shows that the catalytic role of the Fe atom is in distorting the Mg environment. In fact, the Fe atom has a reduced bond distance with its first shell of hydrogen atoms. This implies that a depression around the Fe atom is present. Both the first and the second shell of Mg atoms are thus moved toward the Fe atom (Fig. 3). This displacement lowers the energy needed to free the hydrogen atoms around the Fe atoms.

VI. Conclusion

We have showed that *ab-initio* electronic structure calculations are able to discover atomic level phenomena that are critical for the progress in a technologically relevant issues like the one related to the hydrogen storage.

The availability of an high performance computer like CRESCO allows very accurate simulations of atomic systems in materials science that were not possibile of the critical regions of the simulated system.

Acknowledgments

We gratefully acknowledge drs. G. Bracco, S. Migliori, S. Raia, S. Podda for support in the efficient use of the ENEA CRESCO computational platform.

REFERENCES

[1] A. Montone, J. Grbovic, L. Stamenkovic, A. L. Fiorini, L. Pasquini, E. Bonetti, M. Vittori Antisari, Mat. Sci. Forum **494**, 137, 2005 .

[2] A. Bassetti, E. Bonetti, L. Pasquini, A. Montone, Y. Grbovic, M. Vittori Antisari, Eur. J. Phys. B **43**, 19, 2005.

[3] F. Cleri, M. Celino, A. Montone, E. Bonetti, L. Pasquini, Mat. Sci. Forum **555**, 349, 2007.

[4] T. Schober, Metall. Trans. A **12**, 951, 1981.

[5] R. Car, M. Parrinello, Phys. Rev. Lett. **55**, 2471, 1985.

[6] CPMD, http://www.cpmd.org/, Copyright IBM Corp. 1990 – 2008, Copyright MPI für Festkörperforshung Stuttgart 1997 – 2001.

[7] S. Goedecker, M. Teter, J. Hutter, Phys. Rev. B, **54**, 1703, 1996.

[8] M. Matsubara, M. Celino, P. S. Salmon, C. Massobrio, Solid State Phenomena **139**, 141, 2008.

[9] M. Bortz, B. Bertheville, G. Battger, K. Yvon, J. of Alloys and Comp. **366**, 269, 2004.

[10] S. Giusepponi, M. Celino, F. Cleri, A. Montone, Solid State Phenomena **139**, 23, 2008.

[11] Please visit the website www.cresco.enea.it for more information.

Massimo Celino graduated in Physics at the University "La Sapienza" of Rome (1992). He got his PhD in Physics at the University of Strasbourg (France, 2002). From 1994 he is working at ENEA. His work is focused on Computational Materials Science and Quantum Chemistry. He teaches in Molecular Modeling courses for the University of Camerino.

Simone Giusepponi graduated in Physics at the University of Perugia (2002). He got his PhD in Physics at the University of Camerino (2006). From 2006 he has a ENEA post-doc position in Computational Materials Science.

Applications of Imaging Analysis to Tokamak Fusion Plasma by ENEA-GRID Technologies

M. Chinnici[1], S. Migliori[2], R. De Angelis[3], S. Borioni[1], S. Pierattini[4]

[1]*FIM-INFOPPQ, Casaccia Research Center, Santa Maria di Galeria, Roma, Italy*
marta.chinnici@enea.it
[2]*ENEA-FIM, Enea-Sede, Roma, Italy*
[3]*ENEA, Frascati Research Center, Associazione EURATOM-ENEA sulla Fusione, Italy*
[4]*ENEA- FIM-INFOGER, Bologna Research Center, Bologna, Italy*

Abstract— Today a number of applications in scientific fields (such as medical industry, astronomy, physics, chemistry, forensics, remote sensing, manufacturing, defence) rely upon *images* to store, display, and provide information about the world around us. The challenge to scientists, engineers and business people is to quickly extract valuable information from raw image data. The primary purpose of our work (within CRESCO Project) - i. e. converting images of a nuclear fusion plasma coming from the experiments (*shots*) of Frascati Tokamak Upgrade (FTU) into information by ENEA-GRID infrastructures– is related to such issue. In particular, we use IDL (Interactive Data Language) in order to quickly access image data and to process them. IDL is a high-level programming language that contains an extensive library of image processing and analysis routines.

***Index Te1ms*— Tokamak, IDL, images analysis, Grid.**

I. INTRODUCTION

The global demand for energy continues to grow and there is an immediate need to find new sources of energy. Controlled thermo nuclear fusion offers significant potential advantages as a clear future source of energy in a scenario where there is increasing concern that the emission of green house gases from burning fossil fuels is producing tremendous climatic change by contaminating the environment. Fusion is a clean energy source. In order to effectively use fusion as energy source, it is of pivotal importance to monitor plasma evolution during fusion experiments. Real-Time Imaging analysis is one of the important non destructive, non-invasive, non-perturbative **Frascati Tokamak Upgrade** plasma diagnostics. This contribution (within the **CRESCO** Project) concerns the retrieval of information coming from the images of Tokamak experiments. In particular, we use IDL (Interactive Data Language) in order to analyze images of plasma. The real-time imaging analysis helps to visualize and estimate relevant parameters of plasma phenomenon like run-away electrons or Marfie. We ported a number of applications which analyse and elaborate images coming from the tokamak database. In details, these applications allow *image quality improvement* (noise reduction, contrast enhancement, distortions correction), *automatic classification*

by pattern recognition algorithms and *brightness analysis*, used to detect images which present a characteristic feature (quite recurrent in the plasma) in the brightness distribution. Since FTU image database is rather huge, we used ENEA- GRID in order to achieve efficient performance in terms of ELAPSED and CPU time: benchmark tests we carried out on different platforms with different type of queues show real and meaningful performance improvements in running jobs by opting for this scheduling solution.

Setting

In modern tokamaks visible and infrared video cameras are becoming more and more important to monitor plasma evolution during fusion experiments. The real-time analysis of FTU images performed by IDL applications (*Falsecolor, Database, Volume, Brightzone*) can really provide relevant information to control the plasma and the safety of the machines.

In the last years video cameras have been extensively used in magnetic confinement fusion experiments for both the understanding of the physics and the safety of the operation. Both visible and InfraRed (IR) images can be used not only to monitor the evolution of a plasma discharge but also to evaluate specific parameters, from the determination of impurity radiation to the distribution of power loads on the plasma facing components. Data analysis is normally performed off-line, due to the high amount of information to be processed, making the data acquired by the camera quantitatively useful only for post pulse evaluations. The main difficulty in using visible or infrared images for plasma feedback control is the fact that real-time image processing is challenging and heavy in terms of processing time, especially when complex tasks are required.

At the beginning, the visualization of FTU images has been done under the Videoftu Project. Since FTU image database is rather huge (**4×10^6 Frames**), we used the *multicase submission* with *multicluster queue* to achieve efficient performance in terms of *elapsed time* and CPU time. In order to reduce the run-time of the processes, the route of *multicase processing* has been utilized.

IDL and ENEA-GRID

In **CRESCO** Project, under the task "Development and Integration of the Grid and 3D Graphics" we ported a number of applications which analyse and elaborate the images coming from the tokamak database.

Image and processing analysis of FTU data through IDL applications: *Falsecolor, Database, Volume, Brightzone*. In details, the applications allow *image quality improvement* (noise reduction, contrast enhancement, distortions correction), *automatic classification by pattern recognition algorithms* and *brightness analysis*, used to detect images with a characteristic feature (quite recurrent in the plasma) in the brightness distribution.

An example is the detection of bright toroidal bands (**i.e.** lying in the vessel' s equatorial plane), which precede the onset of regimes of enhanced gas recycling on the wall (a phenomenon known in tokamaks as 'Marfe'), sometimes followed by distructive events. A second example is the identification of bright spots, characterised by typical shapes and localization, which are due to high energy electrons ('runaway electrons'), potentially dangerous for the vacuum chamber. The applications allow a large number of tokamaks images's classification according to specific events and help understanding their correlation with other physical quantities. On the other hand the achievement of event recognition on timescales shorter than those of the evolution of unwanted events, can provide a useful input for the feedback control of plasma operations

***Method**: The experimental evaluation of the algorithm in IDL environment has been performed through the use of the ENEA-GRID infrastructures for the submission and the execution of jobs (**Fig. 1-2**). We used the multicluster queue to achieve efficient performance in terms of elapsed time. Hence, the experimental evaluation of the algorithm in IDL environment has been performed through the use of the ENEA-GRID infrastructures for the submission and the execution of job.*

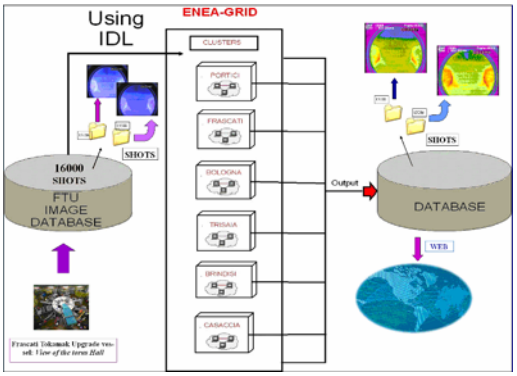


Fig.1 Example of Application: FTU images are input database (**4×10^6 Frames**).

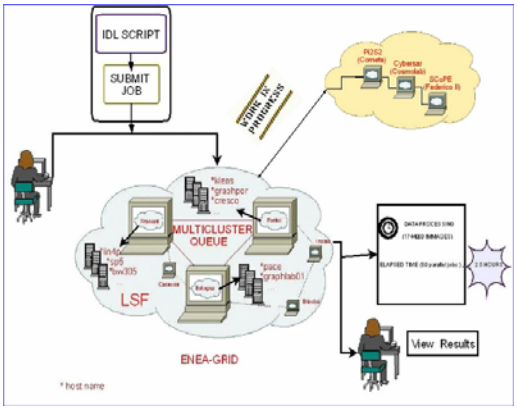


Fig.2 IDL and ENEA-GRID submission: Experimental result with multicluster queue.

*We have used the IDL resources in **ENEA-GRID** infrastructures for the submission and the execution of jobs. In details, we used the multicase submission with multicluster queue that run applications simultaneously on the **6** ENEA-GRID clusters (**Portici, Frascati, Bologna, Trisaia, Brindisi, Casaccia**) in order to achieve efficient performance in terms of ELAPSED and CPU time: benchmark tests we carried out on different platforms with different type of queue show real and meaningful performance improvements in running jobs by opting for this scheduling solution. The images analysis of FTU data through IDL applications are in the output.*

*For example, in order to consider the benefits of multicluster queue, let's consider an FTU experiment where a range of **20 shots** (each shot contains **109 frame**) is produced by a single job:*

- **CPU TIME : ≈ 12 min**
- **CPU TIME (DATABASE) : ≈ 160 hours**

Experimental Tests with distributed run:

- **ELAPSED TIME (10 *parallel jobs*): ≈ 17 hours**
- **ELAPSED TIME (30 *parallel jobs*): ≈ 6.2 hours**

ELAPSED TIME (60 *parallel jobs*): ≈ 3.5 hours

II. Conclusion

Experiments are conducted on ENEA-GRID facilities, on which we submitted the IDL applications in order to analyze images of FTU plasma.

We compare the results of the submission and the execution of jobs with *multicase submission* and without, and show that the utilization of the ENEA-GRID technology is an efficient solution to reduced the run-time required to execute the simulations. In details, we use the multicluster queue to achieve efficient performance in terms of elapsed time.

REFERENCES

[1] R. De Angelis, S. Migliori, S. Borioni, G. Bracco, S. Pierattini, A. Perozziello, **Analysis of images from videocameras in the FTU tokamak,** Review of scientific instruments, Vol. 75, N. 10

[2] http://ftu.frascati.enea.it

[3] **Idl reference guide**, Vol. 1 and Vol. 2

[4] http://cresco/LA1/cresco_sp12_graf3d

[5] M. Chinnici, S. Migliori, R. De Angelis, S. Borioni, S. Pierattini, **Image analysis of a nuclear plasma: Frascati Tokamak Upgrade using IDL and ENEA-GRID technologies,** (Poster) E-Science Conference, Naples, 27-29 May, 2008.

Numerical and Statistical Tools for Image Analysis Based on the Database from Frascati Tokamak Upgrade

M. Chinnici[1], S. Cuomo[2], S. Migliori[3]

[1]*ENEA- FIM-INFOPPQ, Casaccia Research Center, S.Maria di Galeria (Roma), Italy*
marta.chinnici@enea
[2]*UNIVERSITA' FEDERICO II, Dipartimento di Matematica e Applicazioni, Napoli, Italy*
[3]*ENEA- FIM, Enea Sede, Lungotevere Thaon di Revel n. 76 - 00196 Roma, Italy*

Abstract— Imaging analysis is used for analyzing Frascati Tokamak Upgrade plasma. The overall data are organized in an open source portable software system based on a suitable database customized for the FTU. Mathematical and statistical models have been both developed in order to correlate the images with plasma signals and retrieve image information. Some Scientific Computing kernels of the software are implemented in Problem Solving Environments Matlab that provides a comprehensive set of reference-standard algorithms and graphical tools for image processing and analysis. Using Matlab tools the characteristic components of plasma images have been observed.

***Index Terms*— Imaging analysis, Tokamak, Matlab tools, MySql.-Database, ENEA-GRID facilities.**

I.INTRODUCTION

The advent of the internet and extensive digital image libraries have entailed the development of rapid and efficient computer-based image searching and browsing techniques. The term content-based image retrieval refers to automatic recovery of images from a database based on a set of graphic features that qualify the images and that are, loosely speaking, similar to the characteristics of a given query image. This paper (within the **CRESCO Project**) concerns the application of retrieval techniques and analysis of plasma images coming from *Frascati Tokamak Upgrade* (*FTU*). Data from FTU are acquired as movies or single frames. The wide number of information available by visual inspection of the data is often a limitation to real time experimental investigation. It appears that the images from the cameras can help to reconstruct some plasma phenomena and real time analysis could be useful for machine operation. On the other hand, due to the large number of data recorded, visual inspection of the movies is time consuming and often insufficient to correlate the images to other experimental findings. For a profitable use of the signals it is desirable to build specific tools for automatic processing of the images. The

problems increase in complexity and cost in terms of hardware and software resources. In this research field the growing demand for computational resources has been a persistent goal of computational scientists. The challenge is to solve these problems with a computational power that could be achieved inexpensively by collecting "distributed" resources. The main goal is to provide both efficient and low cost computing environment by sharing the computational tools. This paper illustrates the acquisition system and the procedures developed for the processing and analysis of FTU images using **ENEA-GRID** technologies and middleware. In details, data are stored in a suitable database customized for the FTU, to be adopted in mathematical and statistical models in order to correlate the images with plasma signals and retrieve image information.
This contribution is organized as follows. In section 2 we introduce the scenario of FTU. In section 3 we present the "FTUsoftware" methodology and we discuss the results of several retrieval experiments using the "FTUsoftware". Section 4 concludes the paper.

FTU Scenario

Frascati Tokamak Upgrade (FTU) is a compact, high magnetic field tokamak experiment. The objectives of the machine are studying plasma transport, plasma heating and current drive in presence of strong additional radio frequency (RF) heating, and studying plasma profile by means of pellet injection. Plasma discharge lasts about 1.5 s, and in standard operating mode there is an experiment every 20–25 min. The FTU machine—from a control and data acquisition point of view—is composed of four main subsystems: the machine itself (torus, load assembly, cryostat, cooling system); the power supply (two fly-wheel generators, feeding the toroidal magnet and the poloidal windings respectively); RF facility and the diagnostic devices installed on the machine. The RF subsystem is made up of a lower hybrid system (LH: six gyrotrons operating at 8 GHz, total coupled power 2.5 MW), Ion–Bernstein Waves (IBW: three klystrons at 433 MHz, total coupled power 0.4 MW) and an electron cyclotron resonant heating system (ECRH: four gyrotrons at 140 GHz, total power 1.6 MW). The diagnostic devices —generally referred to as 'diagnostics'—include detectors covering the whole range of the electromagnetic spectrum, particle density control devices like DCN laser, or experiments like multi-pellet injection**.**

In this paper, we focus our attention on images of plasma which are observed by wide angle video-cameras placed inside the viewing ***ports*** and close to the plasma edge. The cameras monitor the status of the vacuum vessel and toroidal limiter and give useful data on the occurrence of several events such as arching and flying debris or even major damages such as the detachment of limiter plates. In details, the ports that we considered are: Port_3_hor, Port_5_hor, Port_8_hor (where the suffix "hor" indicates horizontal video-cameras).

"FTUsoftware" methodology

In modern tokamaks visible and infrared video-cameras are becoming more and more important to monitor plasma evolution during fusion experiments. In the last years video-cameras have been extensively used in magnetic confinement fusion experiments for both the understanding of the physics and the safety of the operation. Both visible and Infra Red (IR) images can be used not only to monitor the evolution of a plasma discharge but also to evaluate specific parameters, from the determination of impurity radiation to the distribution of power loads on the plasma facing components. Data analysis is normally performed off-line, due to the high amount of information to be processed, making the data acquired by the camera quantitatively useful only for post pulse evaluations. The main

difficulty in using visible or infrared images for plasma feedback control is the fact that real-time image processing is challenging and heavy in terms of processing time, especially when complex tasks are required.

In order to manage, classify and process the images coming from FTU, we developed a software ("FTUsoftware") that organizes the images in a database and allows to assess the information within the images.

FTUsoftware is implemented by following the software engineering methodology. The main goal is to provide an integrated set of numerical and statistical tools and methods in a framework that is used to structure, plan and control the process of FTU data analysis.

The software is developed in C++ language and the graphic interface is based on Open Qt libraries.

The scientific computing kernel of the software is carried out on Problem Solving Environments Matlab that provides a comprehensive set of reference-standard algorithms and graphical tools for image processing and analysis.

The overall data of the system are stored in a DataBase (named *tokamak*) that is built in MySql **(Fig.1**).

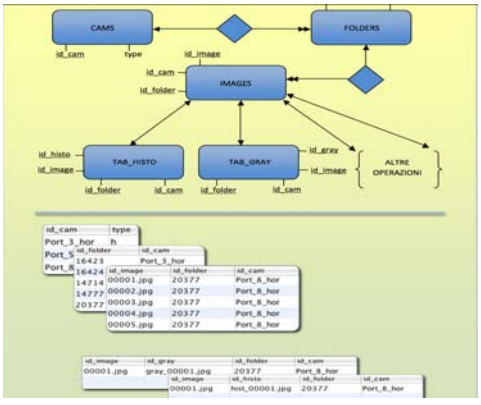


Fig. 1 Scheme of Tokamak Database

In this paper we present a platform-based architecture for developing portable software in order to run it on any machine and any operating system without restrictions.

FTUsoftware usage and ENEA-GRID Middleware

At the beginning it is necessary to set up the Middleware and install the appropriate libraries required to run the executable.

The first step for accessing to images DataBase consists in inserting the IP numerical (e.g. 192.168.1.1) or string (e.g. "localhost") data.

Apart from the IP, the access requires to insert the name of the database, user and password; only authorized users can access to the platform.

In ***the second step*** the system checks the existence of the file "config.conf" that gives us the information about the path of the images (**Fig. 2**).

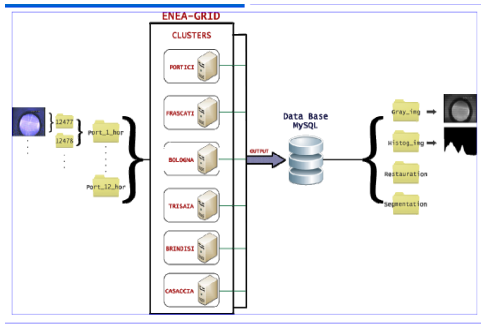


Fig. 2 System pattern.

The presence of the file "Config.conf" guarantees that the FTU images DataBase exists, thus it not necessary to find the images inside the server. On the contrary, if we do not find this file, it is obligatory to insert the path

of the server (**Fig. 3**) – path or machine's folders.

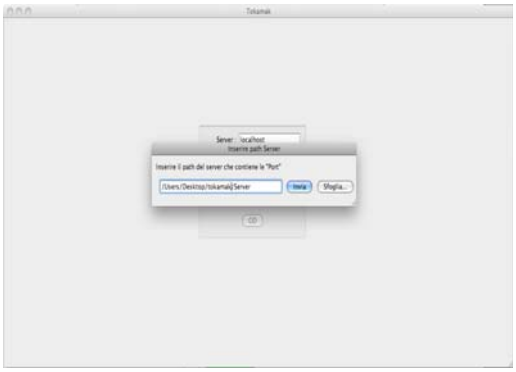


Fig. 3 Using the path to find the images.

Step three. We examine the path; in fact, the path must be structured in a way as showed in **Fig. 4**.

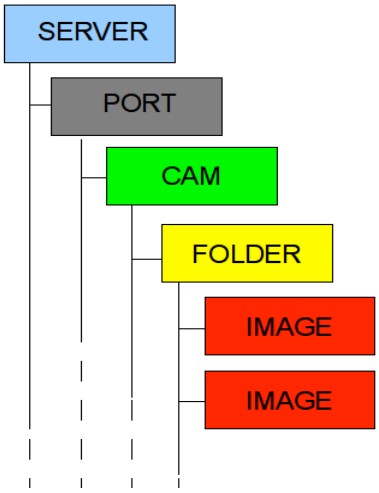


Fig. 4 Prefixed structure of software system.

Step four. FTUsoftware creates folders of our data and if the server has been correctly detected then it fills the folders recursively. Afterwards, FTUsoftware writes the "config.conf" file and the "treeDb.txt" file; the latter arranges the images of Database in order to create dynamical trees.

After these phases, we may enter in the main page (**Fig. 5**) where we can operate on images.

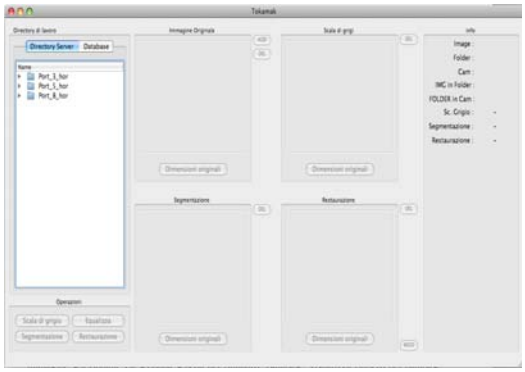


Fig 5 Operating window.

The main page is divided in boxes; the first box on the left includes two sub-folders - Directory Server and Database. The first one contains the tree of the folders (inside the server) from which we pick up one or more images for analysis.

The box has been set in order to show the last image under analysis, even if more images have been utilized. The second sub-folder – Database- shows the tree of the DataBase.

At this point, the selected image is displayed in the box "Immagine originale". When processed (e.g. gray-scale, restoration, segmentation)the image appears in the corresponding box (e.g. "Scala di grigi" etc.). (**Fig. 6**).

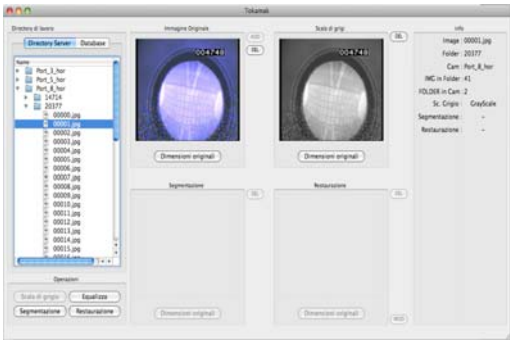


Fig. 6 The boxes show the images under analysis.

It is possible to press the "Dimensioni originali" key to visualize the original dimension of the images (**Fig. 7**).

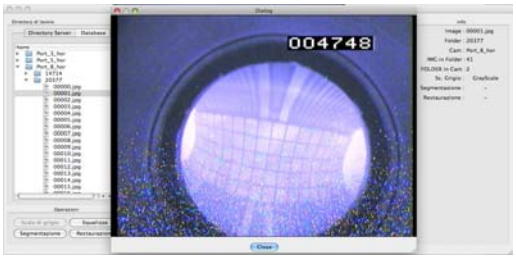


Fig. 7 Original dimensions of an image.

It is possible to insert in or delete from the database an image by pressing "Add" or "Del" keys. Obviously, in order to avoid error, the "Add" key will be active only if the image is not stored in the database, likewise the "Del" key will be active only if the image is already in the database.

In **Fig. 5-6-7**, the first box on the right (**Fig. 8**) gives us the information about the selected image.

Fig. 8 Information on image-data.

II. Conclusion

This paper presents a software facility "FTUsoftware" aimed at analysing FTU images.

The target is to use FTUsoftware in order to automate the processing of FTU images.

The system is integrated in ENEA-GRID infrastructure and it is capable to link Numerical Mathematics with Statistical tools.

FTUsoftware is an efficient, extremely robust and portable software; it allows to draw out and to detect/analyse the images within a database.

Acknowledgments

This work is parts of the CRESCO Project funded by the ENEA (www.cresco.enea.it).

References

[1] Gonzalez , **Digital Image Processing Using Matlab**.

[2] S. Cuomo, La Trasformata di Wavelet Parallela 2D, di un'immagine digitale, CPS-CNR Tech. Rep. TR-2001 - 04, Febbraio 2001

[3] M. Chinnici, S. Migliori, R. De Angelis, S. Borioni, S. Pierattini, **Image analysis of a nuclear plasma: Frascati Tokamak Upgrade using IDL and ENEA-GRID technologies** (poster) - E-Science conference, Nsaples 2008

[4] R. De Angelis, S. Migliori, S. Borioni, G. Bracco, S. Pierattini, A. Perozziello, **Analysis of images from videocameras in the FTU tokamak,** Review of scientific instruments, Vol. 75, N. 10
[5] http://www.cresco.enea.it/LA1/cresco_ sp12_graf3d/
[6] http://tangentsoft.net/mysql++/

The DAME/VO-Neural Infrastructure: an Integrated Data Mining System Support for the Science Community

M. Brescia[1,2], A. Corazza[3], S. Cavuoti[2], G. d'Angelo[2], R. D'Abrusco[2], C. Donalek[4], S.G. Djorgovski[4], N. Deniskina[2], M. Fiore[3], M. Garofalo[2], O. Laurino[2], G. Longo[1,2], A. Mahabal[4], F. Manna[3], A. Nocella[2], B. Skordovski[2]

[1]*INAF-OACN, National Institute of Astrophysics, Naples, Italy*
[2]*Department of Physic Science, University Federico II, Naples, Italy*
[3]*Department of Computer Science, University Federico II, Naples, Italy*
[4]*CALTECH, Californian Institute of Technology, Pasadena, California USA*

***Abstract* — Astronomical data are gathered through a very large number of heterogeneous techniques and stored in very diversified and often incompatible data repositories. Moreover in the e-science environment, it is needed to integrate services across distributed, heterogeneous, dynamic "virtual organizations" formed by different resources within a single enterprise and/or external resource sharing and service provider relationships. The DAME/VO-Neural project, run jointly by the University Federico II, INAF (National Institute of Astrophysics) Astronomical Observatories of Napoli and Trieste and the California Institute of Technology, aims at creating a single, sustainable, distributed e-infrastructure for data mining and exploration in massive data sets, to be offered to the astronomical (but not only) community as a web application. The framework makes use of distributed computing environments (e.g. S.Co.P.E.) and matches the international IVOA standards and requirements. The integration process is technically challenging due to the need of achieving a specific quality of service when running on top of different native platforms. In these terms, the result of the DAME/VO-Neural project effort will be a service-oriented architecture, obtained by using appropriate standards and incorporating Grid paradigms and restful Web services frameworks where needed, that will have as main target the integration of interdisciplinary distributed systems within and across organizational domains.**

***Index Terms* — Grid, SCOPE, Data mining, Virtual Observatory, software engineering.**

I. INTRODUCTION

Modern scientific data (such as in the Astrophysics environment) mainly consist of huge datasets that are gathered by a very large number of techniques and stored in very diversified and often incompatible data repositories. More in general, in the e-science environment, it is considered as a critical and urgent requirement to integrate services across distributed, heterogeneous, dynamic "virtual organizations" formed by different resources within a single enterprise and/or external resource sharing and service provider relationships. One first and important step in this direction has already been undertaken within the Astrophysics community with a set of initiatives that have flourished under the generic name of Virtual Observatory (VObs). The VObs, organized worldwide by means of

the International Virtual Observatory Alliance (IVOA), has defined a set of standards to allow interoperability among different archives and databases in the astrophysical domain, and keeps them updated through the activity of dedicated working groups. One of the first main goals of the IVOA is the federation under common standards of all astronomical archives available worldwide. The concept is that having this meta-archive completed, its exploitation allows a new type of multi-wavelength, multi-epoch science which can only be barely imagined, but also poses unprecedented computing problems. So far, up to now most of the implementation effort for the VObs has concerned the storage, standardization and interoperability of the data together with the computational infrastructures. In particular it has focused on the realization of the low-level tools and on the definition of standards. Our project extends this fundamental target, by integrating it in a service oriented infrastructure, including the implementation of advanced tools for Massive Data Sets (MDS) exploration, soft computing, data mining (DM) and Knowledge Discovery in Databases (KDD).

A. The theoretical domain

From the scientific point of view, the DAME/VO-Neural project arises from the astrophysical domain, where the understanding of the universe beyond the Solar System is based on just a few information carriers: photons in several wavelengths, cosmic rays, neutrinos and gravitational waves. Each of these carriers has it peculiarities and weaknesses from the scientific point of view: they sample different energy ranges, endure different kinds and levels of interference during their cosmic journey (e.g. photons are absorbed while charged Cosmic Rays (CRs) are deflected by magnetic fields), sample different physical phenomena (e.g. thermal, non thermal and stimulated emission mechanisms), and require very different technologies for their detection. So far, the international community needs modern infrastructures for the exploitation of the ever increasing amount of data (of the order of PetaByte/year) produced by the new generation of telescopes and space borne instruments, as well as by numerical simulations of exploding complexity. More in detail, basic astrophysical requirements can be summarized in two items:

- The need of a "federation" of observed and simulated data, by collecting them through several worldwide archives and by defining a series of standards for their formats and access protocols;
- The implementation of innovative computing tools for data exploration, mining and knowledge extraction, user-friendly, scalable and as much as possible automatic;

In other words, the incoming availability of multi-band and multivariate data, from ever more accurate observations of the Universe, introduces the urgency to adopt more powerful soft computing methods to explore from different perspective the MDS. Having such DM tools available, it could be possible to go deeply into the investigation of main astrophysical problems, like:

- Detection and study of the photometric (variable objects) and astrometric (Near Earth Objects or NEO) transients in archival and digital survey data;
- Physical classification of the extra-galactic objects paying special care to the spectroscopic classification of Active Galactic Nuclei (AGN);
- Better knowledge on AGN physical properties;
- Star/Galaxy separation and classification;
- Automatic evaluation of the Point Spread Function (PSF) in frames coming from digital surveys;
- Auto-adaptive integration of

spectroscopic and photometric data, such as the evaluation of photometric redshifts as generalization of feature learning on spectroscopic data;

- Integrated modeling of extremely large telescopes and complex focal plane instruments, through simulation pipelines, covering optical, Finite Element Analysis (FEA), mechanical and control design aspects;

All these topics require powerful, computationally distributed and adaptive tools able to explore, extract and correlate knowledge from multivariate MDS in a multi-dimensional parameter space. The latter results as a typical data mining requirement, dealing with many scientific, social and technological environments.

Concerning the specific astrophysical aspects, the problem, in fact, can be analytically expressed as follows:

Any observed (or simulated) datum defines a point (region) in a subset of R^N, such as:

- R.A. and DEC;
- time and λ;
- experimental setup (i.e. spatial and/or spectral resolution, limiting magnitude, brightness, etc.);
- fluxes;
- polarization;
- spectral response of the instrument;
- PSF;

Every time a new technology enlarges the parameter space or allows a better sampling of it, new discoveries are bound to take place.

So far, the scientific exploitation of a multi-band (D bands), multi-epoch (K epochs) universe implies to search for patterns and trends among N points in a DxK dimensional parameter space, where $N > 10^9$, $D >> 100$, $K > 10$.

The problem also requires a multi-disciplinary approach, covering aspects belonging to Astronomy, Physics, Biology, Information Technology, Artificial Intelligence, Engineering and Statistics environments. In this sense, experimental astronomy has in practice become a three players game, made by:

- Astronomers: theory, data, understanding, discoveries, structure, biases;
- Statisticians: evaluation of data, validation, analysis, dimensional reduction, models;
- Computer Scientists: implementation of infrastructures, archives and databases, information retrieval, middleware, scalable tools, OOP;

B. Data Exploration

The requirements coming from above theoretical scenario involves a data exploration conceptual domain based on the approach summarized in Fig. 1.

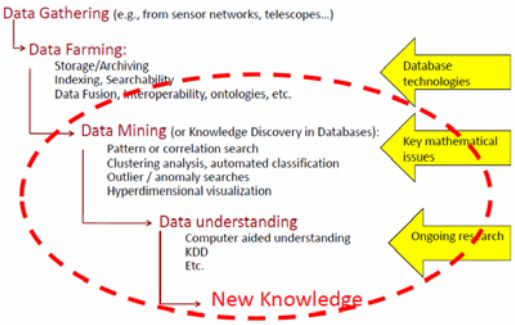


Fig. 1 – Data exploration conceptual domain

In the last decades data exploration research has been driven by the advance of database technology and the creation of huge datasets. The growth of "visual and/or numerical analytics", has emphasized the danger of information overload and the need to harness new technology, such as high resolution and multivariate analysis systems, in order to maintain overview and control of the data. DAME/VO-Neural group have always stressed the need to organize in an homogeneous way

both syntax and semantics of data manipulation, analysis and reduction, with the aim at the creation ontologies to ease communication between scientists using a distributed computing infrastructure such as the S.Co.P.E. GRID, representing environment architecture which is main deployment target of DAME/VO-Neural Project. Despite these advances, the problem of finding suitable methods for the exploration of large datasets remains still formidable. The emphasis is on methods of bridging the gap between accurate representations and mining of the data and the capabilities of current technology and users. The analytical methods based partially on statistical random choices (crossover/mutation) and on knowledge experience acquired (supervised and/or unsupervised adaptive learning) could realistically achieve the discovery of hidden laws behind focused phenomena, often based on nature laws, therefore the simplest.

During the R&D phase of our project, aimed at define and characterize rules, targets, ontologies, semantics and syntax standards, the following functional breakdown structure has been derived:

- Exploration
- Dimensional Reduction
- Regression
- Clustering
- Classification
- Visualization

It provides a taxonomy between possible data exploration modes, made available by our infrastructure as data mining experiment typology (use case). In order to accelerate the infrastructure prototype development, in the first implementation phase it was decided to reduce this scenario, by focusing the attention on Classification and Regression functionalities only.

1) Data Exploration: Classification

Statistical classification is a procedure in which individual items are placed into groups based on quantitative information on one or more features inherent to the items (referred to as features) and based on a training set of previously labelled items. A classifier is a system that performs a mapping from a feature space X to a set of labels Y. Basically a classifier assigns a pre-defined class label to a sample. Formally, the problem can be stated as follows: given training data {(x_1,y_1),...,(x_n, y_n)}, (where x_i are vectors), a classifier h:X->Y maps an object x ε X to its classification label y ε Y.

Different classification problems could arise:

a) crispy classification: given an input pattern x (vector) the classifier returns its computed label y (scalar).

b) probabilistic classification: given an input pattern x (vector) the classifier returns a vector y which contains the probability of y_i to be the "right" label for x. In other words in this case we seek, for each input vector, the probability of its membership to the class y_i (for each y_i).

Both cases may be applied to both "two-class" and "multi-class" classification.

2) Data Exploration: Regression

We will define Regression as the supervised search for a mapping from a domain in R^n to a domain in R^m. One can distinguish between two different types of regression:

a) data table statistical correlation: the user tries to find a mapping without any prior assumption on the functional form of the data distribution. Machine learning algorithms are well suited for this kind of regression;

b) function fitting: with curve fitting the user tries to validate the hypothesis, suggested by some theoretical framework, that the data distribution follows a well defined, and known, function;

A regression system performs a mapping from a parameter space X to a target space Y. Formally, the problem can be stated as follows: given training data {(x_1,y_1),...,(x_n, y_n)} (where x_i are vectors) a regression operator h:X->Y maps an object x ε X to its value y ε Y.

C. *The Project Infrastructure*

By taking into account both theoretical and conceptual domains, we designed an infrastructure based on the following skill features:

- ✓ Object Oriented Programming;
- ✓ Internal standards and protocols (VO, XML);
- ✓ Java language (almost generic for data mining models);
- ✓ User/Session Registry DataBase Management System (MySQL);
- ✓ Web-based User I/O;
- ✓ Service-Oriented Web Application and Restful Web Service Technology, Servlet (Web Server applets);
- ✓ Plugin-based Modularity (easy to be integrated/modified) for data mining models;
- ✓ Hardware independent through GRID driver;
- ✓ Data conversion and manipulation support (ASCII, FITS, CSV, VOTable);

In the following sections, each of the infrastructure components is described.

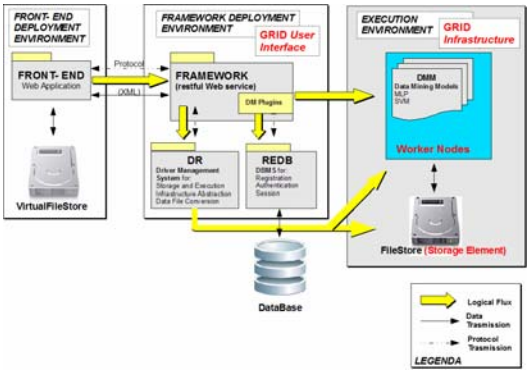


Fig. 2 – DAME/VO-Neural Infrastructure Architecture

1) *Front End Component*

The Front End (FE) represents the main interface between the infrastructure core and the external world (user who wants to submit own experiments). It is structured as a Service-Oriented (SO) tool implemented by a WEB application, whose scope is to furnish a I/O data and functionalities computing interface to the user. This choice has been driven taking into account following requirements:

- To provide a remote distributed application, whose internal mechanisms are totally hidden to the final user;
- To avoid any HW/SW client dependence. It is possible to use any kind of public web browser to interact with the WEB application in a dynamical (context-dependent) way;
- To implement an easy and safe security policy, handled by the Servlet (Web Server applet);
- To obtain an easy bug-free and maintainable application;
- To provide a flexible instrument easy to be updated and expanded;

A registered and authenticated user can access the FE via browser in order to fill in the login form obtained from the previous registration. In case of deployment of the infrastructure on a GRID platform, a robot certificate system is automatically provided in a transparent way to

the final user. The GUI of the FE permits the navigation of the user between his own working sessions (term introduced to classify user experiments, i.e. associations between chosen datasets, data mining model and functionality), the remote upload of user data files, their interactive manipulation (preparation of data sets for the specific data mining experiment) and storing for future experiments. The FE also provides the interactive/asynchronous check of experiment status, the download of final or partial results and the data mining model-dependent output graphics and charts eventually selected by the user.

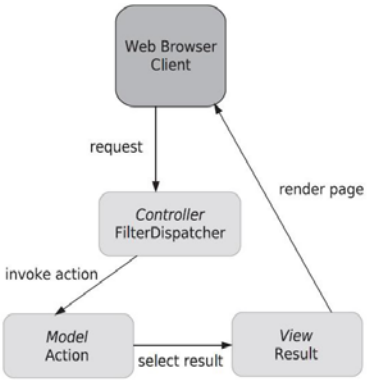


Fig. 3 – The FE MVC pattern implementation

Any user request is internally processed through the XML-based bi-directional communication between FE and the core component (FrameWork).

The FE internal mechanism is based on the Model-View-Controller (MVC) pattern that provides a separation of concerns that applies well to WEB applications. Separation of concerns allows us to manage the complexity of large software systems by dividing them into high-level components.

The MVC design pattern identifies three distinct concerns: *model*, *view*, and *controller*. In the FE, these are implemented by the Action, Result, and Filter Dispatcher respectively (Fig. 3).

2) FrameWork Component

The DAME/VO-Neural Suite is composed of several components which relies on a common infrastructure to perform their task. This infrastructure is basically provided by the FrameWork (FW) component. Nonetheless this component, while providing this common platform, represents the core of the project, since it is the component who actually implements, configures and launches the data mining experiments. In our analysis of the FW component, every operation can be considered atomic, so the natural choice is to implement the web service in the RESTful architectural style. A RESTful Web Service is a client\server architecture in which the web services are viewed as resources and are identified by their URLs. A client that wants to access to a resource can use common http methods like get and post on the URL that identifies the desired resource. A RESTful environment is completely stateless, that is every interaction is atomic. The main advantage of this style is its great simplicity, which allows a fast deployment and simple interactions with the external world.

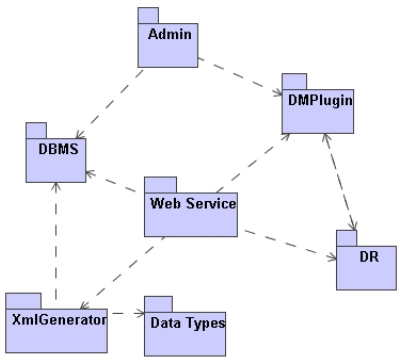


Fig. 4 – The FW class diagram

We identified Servlets (namely an object which accept a request from a client and creates a response based on that request) to implement the RESTful Web Service. For every resource there will be a Servlet associated to it, which overrides the standard methods to implement the operation associated to that resource. Furthermore, the FW

component needs to implement an Admin interface. An Admin needs to interact with the FW to add or remove plug-ins (DMMPlugin) and possibly to request statistical information like number of users and number of experiments. Moreover, one of the most critical factors of the FW is the interaction of a newly created experiment with the GRID platform. The FW component needs to create and configure the DMPlugin associated to the experiment. We called DMPlugin a module encapsulating experiment and associated Data Mining (DM) configuration and working flow methods. After such a plug-in is configured it needs to run the experiment by calling the run method of the plug-in itself. In GRID, as known, the process needs to migrate on a Worker Node (WN). To implement this migration we've chosen to serialize the DMPlugin in a file. Serialization is a process of converting an object into a sequence of bits so that it can be stored on a storage medium. To manage the interactions between the FE and the FW the adopted standard protocol, in some operations like the creation of a new experiment, is based on XML-type files. The FW and the DMPlugin have methods to create and parse XML files to apply user (FE) requests.

3) *REgistry & DB Component*

The Registry & DataBase (REDB) is the Suite component including the DataBase Management System (DBMS) component and its Interface. The REDB handles user's registration, authentication, working sessions, experiments and files; its DBMS manages data and relationships about Users, Sessions, Experiments, Functionalities and Files. The component architecture is based on the following entities: a relational DBMS composed by a DB Server and a text interface Client that supports the SQL syntax, a DBC Driver, a DBC API and the REDB Interface.

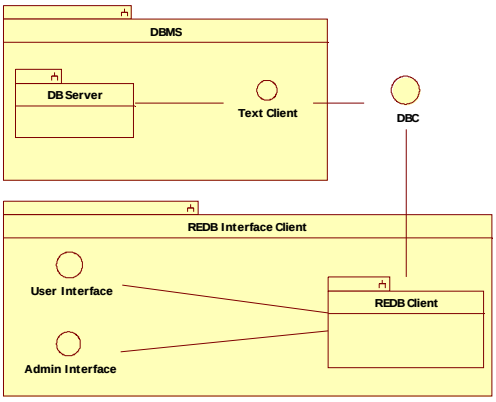


Fig. 5 – The REDB architecture

The component architecture is based on the following technologies: MySQL, Java as interface programming language, JDBC Driver and its API (Fig. 6).

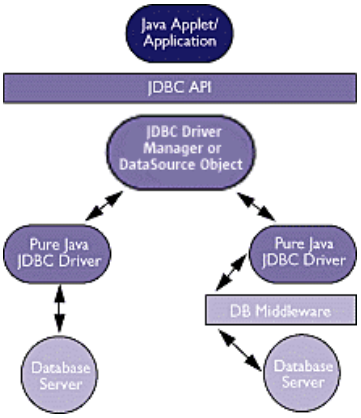


Fig. 6 – The JDBC driver architecture

4) *DRiver Component*

To separate FW functional requirements from their implementation issues, it was necessary to create a library of methods called DRiver (DR) Management System. The DR is the component used by the FW to manage the processing environment. It implements the low-level interface with computational environment, in order to permit the FW implementation, through the specific drivers, on different platforms (such as Stand-Alone or GRID). In other words, the DR is used to implement the proper access to the platform-

dependent resources required by all the specific use cases and functionalities of the Suite. The experiment I/O data basically consist in the user input and suite output. In order to avoid multiple deployment of these data on several components, it was decided to provide a file storage system (SE or Storage Element on GRID), hosting real data files, handled directly by the FW through the DR component methods. This component includes also a library of data file format (FITS, ASCII, CSV and VOTable) translating methods, used by the FW depending on the specific DM model data format supported.

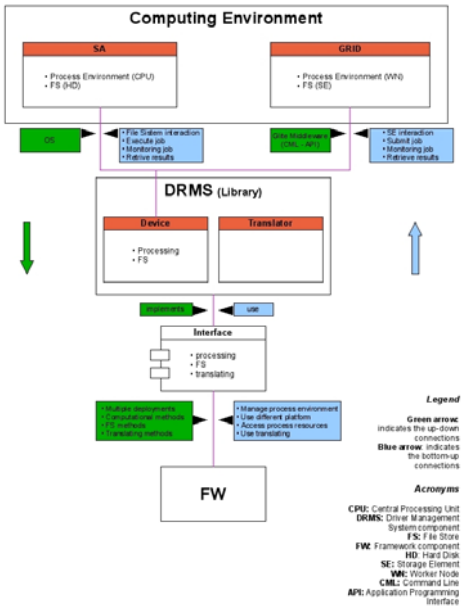


Fig. 6 – The DR component architecture

5) Data Mining Models Component

The DMM is the component that implements the DM models and related wrappers code and their use cases.

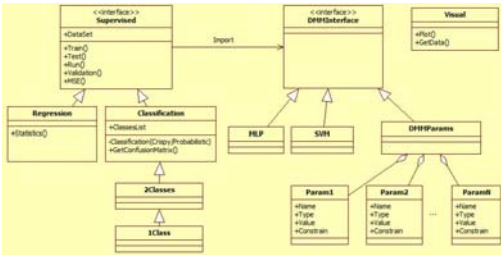


Fig. 7 – The DMM component architecture

Its main features can be summarized as follows:

1. Implementation oriented to the functionalities (i.e. classes of functionalities, such as Classification and Regression);
2. Possibility of use functionalities with more than one model without duplicate code (Pattern Bridge as standard design pattern);
3. Separation between Supervised and Unsupervised models;
4. A common interface for all the models (by means of a specific class rendering typical data mining self-adaptive parameters, called DMMParams);

In the first release, the DMM implements MultiLayer Perceptron (MLP) ([9]) and Support Vector Machine (SVM) ([10]), as supervised data mining models, while Self Organizing Maps (SOM) ([9]) is the unsupervised model foreseen to be implemented in the next Suite release.

II. Conclusion

The DAME/VO-Neural project comes out as an astrophysical data exploration tool, originating from the very simple consideration that, with data obtained by the new generation of instruments, we have reached the physical limit of observations (single photon counting) at almost all wavelengths. Hence, our opportunity to gain new insights on the Universe will depend mainly on the capability

to recognize patterns or trends in the parameter space, which are not limited to the 3-D human visualization, from very large multi-wavelength and multi-technique data sets. Moreover, this approach (and of course the proposed technique) can be easily and widely applied to other scientific, social, industrial and technological scenarios. Our project has recently passed the R&D phase ([1], [2]), de facto entering in the implementation step and by performing in parallel the scientific testing with first infrastructure prototype, accessible, after a simple authentication procedure, through the official project website address (http://voneural.na.infn.it/). First scientific test results ([5], [6], [8]), confirm the goodness of the theoretical choice, of the data mining model design and technological strategy. This demonstrates and encourages the winning approach to integrate VObs experience, massive data exploration, Knowledge Database Discovery, GRID paradigm, Soft Computing, Service-Oriented Information Technology in a unique infrastructure for the entire community.

ACKNOWLEDGMENTS

The authors would like to thank VOTECH and S.Co.P.E. projects for their fundamental support, together with all DAME/VO-Neural team members, contributors and collaborators for their effort employed to reach the successful goal. The project was funded by the Italian Ministry of Foreign Affairs (MAE) through a bilateral Italy-USA agreement.

REFERENCES

[1] M. Brescia, S. Cavuoti, G. d'Angelo, R. D'Abrusco, N. Deniskina, M. Garofalo, O. Laurino, G. Longo, A. Nocella, B. Skordovski, "The VO-Neural project: recent developments and some applications", Contributed, Data Centre Alliance Workshops: GRID and the Virtual Observatory, Munich, to appear in Mem. SAIt , arXiv:0806.1006, April 9-11 2008;

[2] M. Brescia, S. Cavuoti, G. d'Angelo, R. D'Abrusco, C. Donalek, N. Deniskina, O. Laurino, G. Longo, "Astrophysics in S.Co.P.E.", to appear in Mem. SAIt, (Submitted on 7 Jul 2008), http://it.arxiv.org/abs/0807.0967

[3] M. Brescia, G. Longo et al., "Neural Net aided detection of astronomical periodicities in geologic records", Earth and Planetary Science Letters, Elsevier Netherland, March 1996, 1996E&PSL.139...33B, Vol. 139 / 1-2, pp. 33-45;

[4] M. Brescia et al., "A Neural Tool for Ground-Based Telescope Tracking control", AIIA NOTIZIE, periodico dell'Associazione Italiana per l'Intelligenza Artificiale, Anno XVI, December 2003, N° 4, pp. 57-65;

[5] R. D'Abrusco, A. Staiano, G. Longo, M. Brescia, M. Paolillo, E. De Filippis, R. Tagliaferri, "Mining the SDSS Archive I. Photometric redshifts in the nearby Universe", The Astrophysical Journal, 2007 July 10, 663: 752-764;

[6] R. D'Abrusco, G. Longo, A. Staiano, E. De Filippis, M. Paolillo, M. Brescia, "Step towards a map of the nearby universe", Nuclear Physics Proceedings Supplements, 2007, ISSN 0920-5632, vol. 168, 299-301;

[7] N. Deniskina, M. Brescia, S. Cavuoti, G. d'Angelo, O. Laurino, G. Longo, "GRID-Launcher v.1.0", Contributed, Data Centre Alliance Workshops: GRID and the Virtual Observatory, Munich, to appear in Mem. SAIt , arXiv:0806.1144, April 9-11 2008;

[8] O. Laurino, R. D'Abrusco, M. Brescia, S Cavuoti, A. Corazza, G. d'Angelo, C. Donalek, S. G. Djorgovski, N. Deniskina, M. Fiore, M. Garofalo, G. Longo, A. Mahabal, F. Manna, A. Nocella, B. Skordovski, "A web application for photometric redshifts evaluation", to appear on Proceedings of Workshop finale dei Progetti GRID del PON "Ricerca" 2000-2006 – Avviso 1575 – Catania, 2009;

[9] C. M. Bishop, "Neural Networks for Pattern Recognition", Oxford University Press (1995);

[10] C.C. Chang, C. J. Lin, "LIBSVM a library for Support Vector Machines";

M. Brescia Astronomer researcher at INAF-OACN (Astronomical Observatory of Capodimonte of Naples) and member of (IAU) International Astronomical Union. Degree in Computer Science, specialized in astrophysical technology, artificial intelligence and data mining software engineering. Teacher of Computer Architecture at Computer Science Department and Astronomical Technologies at the Physics Department of the University Federico II, Naples, Italy.

A**. Corazza** assistant professor at the University Federico II in Naples, Italy. Before,

she worked at the ITC-irst (nowadays FBK) in Trento and at the University of Milan. Her research interests focus on statistical approaches to natural language processing, bioinformatics, and information retrieval.

Many-Body Systems HPC Applications on the PI2S2 Grid

D. Gambacurta[1,2,3] and F.Catara[2,3]
[1]*Consorzio COMETA, Via Santa Sofia 64, 95123, Catania, Italy, danilo.gambacurta@ct.infn.it*
[2]*Dipartimento di Fisica e Astronomia, Via Santa Sofia 64, I-95123 Catania, Italy*
[3]*Istituto Nazionale di Fisica Nucleare, Sezione di Catania, Via S.Sofia 64, I-95123 Catania, Italy*

***Abstract*—In the present work we show some results obtained by using the PI2S2 infrastructure of COMETA. First of all we discuss some features of SLEPC, a free software library for the solution of large sparse eigenvalue problems. The use of parallel eigensolvers, in which data are distributed across several processors, allows both to increase the available memory and to reduce the computation time. After that, some calculations of collective excitations in Metal Clusters within extended RPA approaches are discussed. Collective Electronic modes of Metal Clusters within (i) an extended RPA approach and (ii) the Second RPA framework are studied. In the first case a better agreement with experimental results with respect to standard RPA is found while in the second case, some problematic aspects of SRPA are underlined and analyzed.**

I. Introduction

THE quantum many-body problem is a still open challenge that has not yet received a satisfactory complete solution. Indeed, the physics of interacting particles is a very complicated subject, because the motion of each particle depends on the position of all the others. In other words, the motion of the particles become correlated. The approach to the many-body problem can be divided on the basis of the evaluation of a hierarchy of correlations. First comes the one-body or mean-field theory. Second the two-body correlations, and so on. From a numerical point of view, the more one goes up this hierarchy and usually the more expensive and heavy calculations have to be performed. The Grid infrastructure is a particularly well suited tool for their implementation. In this paper we briefly present some results obtained by using the COMETA Grid infrastructure to implement and run our applications.

The eigenvalue problem is one of the most recurring numerical tasks in scientific disciplines. For example, in nuclear physics, Shell Model or RPA calculations, that are fundamental tools for the study of nuclear structure, consist, from a numerically point of view, in solving an eigenvalue problem. In the physics of many-body systems the matrix to be diagonalized is typically the representation of the Hamiltonian operator H in some finite basis of orthonormal functions which is assumed to be large enough to give useful results. The quality of the basis set can be checked by comparing the results of calculations using progressively larger basis sets. However, by increasing the dimension of the basis, the calculations become extremely difficult, because matrix dimensions rapidly increase. Thus, standard direct methods of diagonalization are often not suited and one has to resort to other techniques, as for example iterative methods. In Section II we test the SLEPC library [1], a free software library for the solution of large sparse eigenvalue problems on parallel computers, paying particular attention to its parallel efficiency.

Section III is devoted to the analysis of

electronic properties of Metal Clusters within the Random Phase Approximation (RPA) and two its extensions. RPA calculations are usually carried out in order to improve the standard mean field description of a physical system. In this scheme, excited states are described as a superposition of 1 particle - 1 hole and 1 hole - 1 particle excitations built on top of the ground state, that is assumed to be correlated. However, RPA presents some limits and many efforts are still done by the scientific community in order to overcome them by improving the RPA description of ground state correlations. We have thus developed an improved RPA approach in order to take into account correlations in a self-consistent way. In particular we have analyzed the dipole plasmon modes in Metal Clusters and we have found a good agreement with experimental results, better than standard RPA. In Section III-A the main results are presented.

In Section III-B we show some results obtained within the Second RPA. This is a natural extension of RPA and consists essentially in the introduction of more general excitation operators, containing also 2 particle - 2 hole terms, in order to give a better description of double excitations. Both in RPA and SRPA one has to resort to the Quasi Boson Approximation (QBA) that amounts to replace, in the evaluation of the matrix elements of RPA (SRPA) equations, the correlated state, which is assumed to exist and to be the vacuum of the phonon RPA (SRPA) operators, with the uncorrelated Hartree-Fock one. However, it has been argued that its use is even more problematic in SRPA than in RPA. Thus we decided to analyse merits and limitations of SRPA by applying it to the study of the vibrational spectrum of metal clusters. We have found that within SRPA one gets very strong modifications of the RPA excitation spectrum, the energy being pushed down, also for the collective states whose description within RPA is widely accepted as satisfactory. In Ref. [3] a detailed analysis of this behaviour allows us to conclude that a better treatment of short range correlations, along the path of extended RPA and SRPA approaches, as done for example in [2], [4], is necessary.

II. SLEPC Parallel EigenSolvers

The eigenvalue problem is one of the most recurring numerical tasks in scientific disciplines. Very often, standard direct methods of diagonalization are not suited and one has to resort to other techniques, as for example iterative methods. The main critical points are generally the amount of needed memory and the time spent in the diagonalization process. The use of modern computer resources and parallel eigensolvers, in which data are distributed across several processors, allows both to increase the available memory and to reduce the computation time. In the following we show some results obtained by using SLEPC library [1]. It allows to treat standard and generalized eigenvalue problems, for both Hermitian and non-Hermitian matrices, and to select different eigensolvers (Arnoldi, Lanczos, Krylov-Schur, etc..). We will focus our attention to the case of a matrix whose dimension is $D \sim 3 \times 10^6$ and whose structure is such that the memory needed to allocate the non-zero matrix elements (about 5GB) exceeds the physical memory limit (4GB) of a single node. In this case, the use of multiple CPUs allows to carry out a task otherwise not (easily) affordable. In the upper panel of Fig. 1 the dependence of the CPU time for different numbers of processors is shown for the 5 lowest requested eigenvalues while in the lower one the speedup is plotted. The Krylov-Schur algorithm has been used. The convergence tolerance has been chosen equal to 10^{-6}. In the upper panel of the figure, we can see that increasing the number of CPUs, a reduction of the consumed time is obtained. A natural indicator of parallel scalability is the speedup. It is defined as the CPU time to solve a given task as performed on a single CPU, divided by that of the same task performed on multiple CPUs. It shows whether CPU time scales linearly with the number of used CPUs. Since the serial job can not be performed, we assume as serial CPU time the one obtained with 5 CPU divided by 5. In the

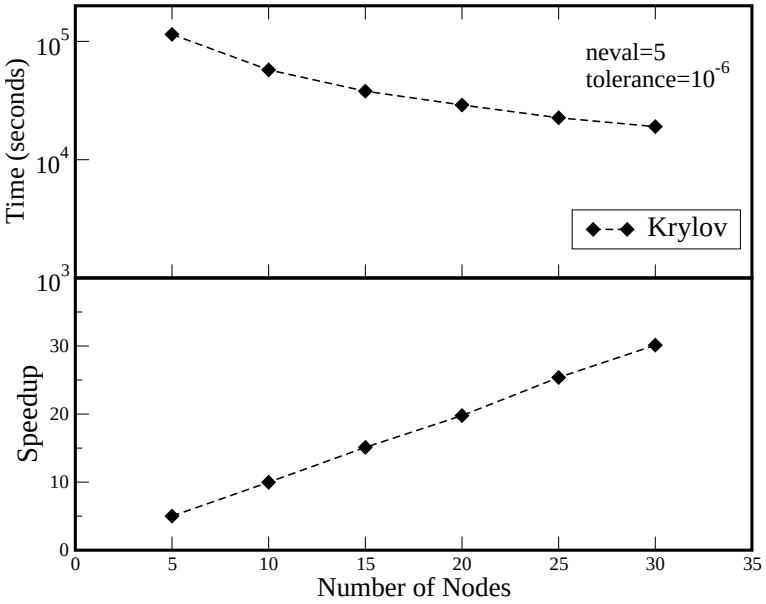


Fig. 1. In the upper panel of the figure the CPU time as a function of the number of nodes. The results are obtained with Krylov-Schur algorithm. The number of requested eigenvalues is "neval"=5 and the convergence tolerance has been chosen equal to 10^{-6}. In the lower panel speedup as a function of the number of nodes is plotted.

lower panel of the figure we observe that also a very good scalability is obtained.

III. Collective Excitations in metal clusters

One of the common features characterizing finite-size many-body systems is the existence of collective modes of excitations. In metal clusters, the dipole plasmon [5] is the analogue of the nuclear dipole GR [6] and is interpreted as due to the collective vibration of the electrons against the positive ions. In the following we will show some results about collective excitations in metal clusters by using two different extensions of the random-phase approximation (RPA) aimed to overcome some of its limits.

A. Extended RPA description

In [4] we presented an extension of RPA, that we will call ERPA, aimed to give a better description of ground state correlations in order to overcome the Quasi Boson Approximation, a well known limit of standard RPA. In the present paper we will only show the main results, for more details about the method and the procedure see Ref. [4]. We only stress that ERPA equations are non linear and have to be solved iteratively, requiring a big computational effort. However, the use of the COMETA Grid allowed us to carry out a systematic study of several metal clusters.

In Fig. 2 we illustrate the spin $S = 0$ dipole strength distributions for the alkali clusters Na_N with N=8, 20, 40, and 58. In order to make clearer the comparison, the discrete lines of RPA and ERPA spectra are folded with a Lorentzian function. In both cases an artificial width $\Gamma = 0.1eV$ has been used. Solid lines refer to calculations performed within the present approach whereas dashed lines refer to standard RPA calculations. In all cases one observes that

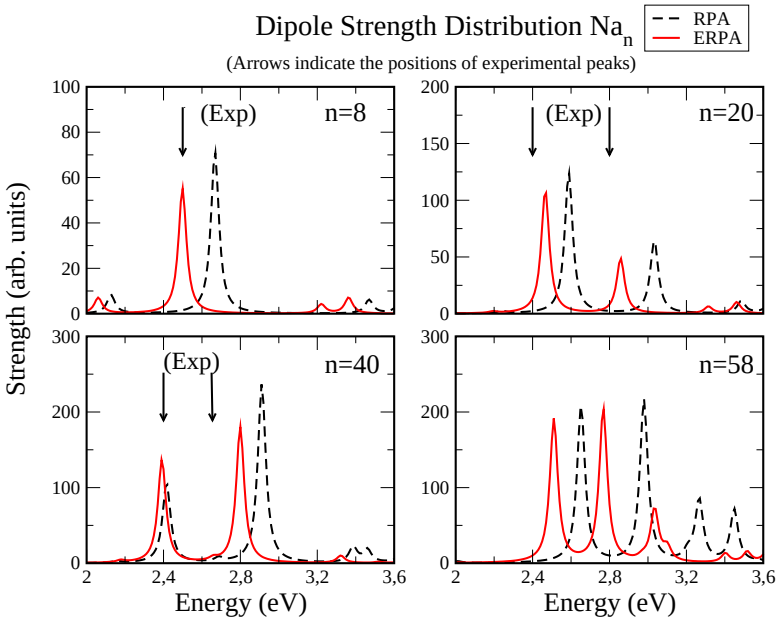


Fig. 2. (Color online) Spin=0 dipole strength for Na_N metal cluster with N =8, 20, 40 and 58. Solid (red) line refers to calculations performed within our extended RPA approach (ERPA), while the dashed (black) line refers to standard RPA calculations

the basic structure of the distributions remains essentially unchanged. However, an overall shift of the distributions to lower energies, as compared to the RPA results, is present. A comparison with the experimental photoabsorption cross sections [7], [8] turns out particularly useful. In the case of Na_8 , the experimental distribution exhibits a broad peak centered at about 2.5 eV, which is well in agreement with the plot of Fig. 2. A similar good agreement is observed in the case of Na_{20} where the experimental distribution shows two broad peaks centered at about 2.4 and 2.8 eV, the first peak with a strength more than double the second one. Both in the case of Na_8 and Na_{20}, then, the shift produced by the present calculations with respect to RPA ones improves the fit with the experimental data. The situation is more complicated in the case of Na_{40}. Here, in fact, the experimental distribution exhibits two broad peaks at about 2.4 and 2.65 eV and the first one is higher than the second one. As we can observe, RPA predicts the first peak at an energy of about 2.45 eV, very close to the experimental value, and the second one much higher, at about 2.97 eV. Furthermore, the height of the second peak is about twice that of the first one. In ERPA, both peaks are shifted to lower energies, at about 2.39 eV and 2.79 eV, in agreement with the experimental values. Also the structure of the distribution changes and a redistribution of the strength, qualitatively in better agreement with the experimental results, is observed. In particular we have an increasing of the height of the first peak and a lowering of the second one. However, the strength distribution remains quite different from the experimental one. To our knowledge, no experimental data exist at present for Na_{58}.tex

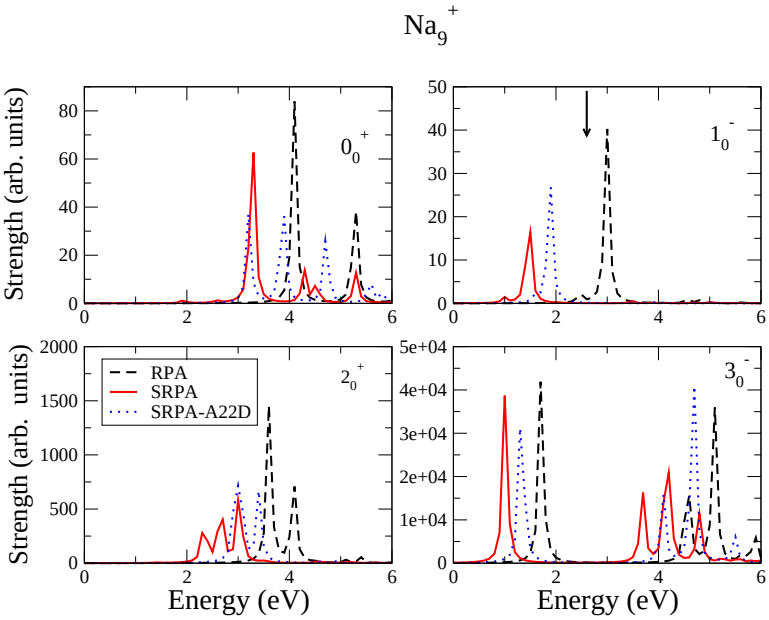


Fig. 3. (Color online) Natural parity $L = 0, 1$ (upper panel) and $L = 2, 3$ (lower panel) spin=0 multipole strength distributions for Na_9^+ metal cluster are shown. Solid (red) lines refer to SRPA results, while the dashed (black) ones refer to RPA calculations. SRPA-A22D dotted (blue) lines show the SRPA results when the diagonal approximation (see the text) is used. The arrow roughly indicates the position of the experimental dipole plasmon peak.

B. Second RPA

Second RPA (SRPA) is a natural extension of RPA and consists essentially in the introduction of two particle-two hole ($2p-2h$) excitations, in addition to the one particle-one hole ($1p-1h$) ones already taken into account at RPA level. The derivation of SRPA is also based on QBA and it has been argued that in this case its use is even more problematic than in RPA. This point has been recently analyzed in [2], where an approach to go beyond QBA was presented and applied to a 3 level Lipkin model. However, such analysis is limited by the fact that the model is very schematic and contains several parameters. In particular, the relative strengths of the terms of the interaction are completely arbitrary. Therefore, one somehow fixes such relative strengths and studies the evolution of the results as a function of an overall interaction strength. By doing so, however, it remains open the question of how much the quality of the agreement with exact results depends on the chosen relative weights.

In the following we analyze merits and limitations of SRPA for the study of the vibrational spectrum of metal clusters. We make use of the complete SRPA scheme, including all kinds of coupling within the $1p-1h$ and $2p-2h$ configurations as they come out from the underlying two body interaction. To our knowledge, this has never been done before since current versions of SRPA are based on non-interacting $2p-2h$ configurations and only interactions of $2p-2h$ with $1p-1h$ have been explicitly taken into account. Indeed, having in mind to use SRPA as a tool for studying the spreading width of the RPA ("one-phonon") collective states, in the calculations done till now the coupling of the $2p-2h$ configurations among themselves has been completely neglected. As we will see,

this approximation, that we will call "diagonal approximation", does not account fully for the energy shifts of the "one-phonon"states. In addition to that, of course the study of the spectrum corresponding to double excitations ("two-phonon" states) necessarily requires to take into account those neglect ed couplings. The name diagonal approximation derives from the fact that in the matrix describing the coupling of the $2p-2h$ configurations among themselves only the diagonal unperturbed elements are taken into account.

Here we will only show the main results, for more details about the method and the procedure see Ref. [3] In Fig. 3 we plot the strength distributions for the multipole operator $F^{(\lambda)}(r) = r^{\lambda}Y_{\lambda 0}$ for the Na_9^+ metal cluster. Also in this case, in order to make clearer the comparison, the discrete lines of RPA and SRPA spectra are folded with a Lorentzian function. In both cases an artificial width $\Gamma = 0.1eV$ has been used. In the upper and lower panel of Fig 3, we show, respectively, the $L = 0, 1$ and $L = 2, 3$ multipole strength distributions with spin $S = 0$ for Na_9^+ metal cluster. Solid (red) lines refer to SRPA results, while the dashed (black) ones refer to those obtained within RPA. With the dotted (blue) lines we indicate the SRPA results when the diagonal approximation is used.

We see that going from RPA to SRPA the strength distributions are shifted to lower energies, especially in the dipole case. The results shown in Fig. 3 refer to the lower part of the excitation spectrum, which is mainly composed by $1p-1h$ configurations. However, even for such states the inclusion of the coupling among $2p-2h$ configurations leads to a further shift down. Let us look in some detail to the collective dipole $S = 0$ state, which in RPA is located at 2.98 eV, and that experimentally is found at about 2.60 eV [8]. Within the approximated SRPA, that is when only the coupling of $1p-1h$ to $2p-2h$ configurations is taken into account, the energy moves to 1.88 eV, while in the full SRPA one gets 1.47 eV.

In order to derive the equations of motion, both in RPA and SRPA, use is made of the QBA. However, the replacement of the correlated ground state with the HF one is justified only if ground state correlations are not too strong. In RPA the Y_{ph}^{ν} amplitudes are a measure of these correlations and thus, looking at their behaviour going from RPA to SRPA could be useful in order to have more information about the adequacy of QBA. We have seen that going from RPA to SRPA an increase of these quantities is observed. Since, as mentioned above, the use of QBA is justified only when the HF state does not differ very much from the correlated one, the larger deviations found in SRPA suggest that this approximation could be more severe than in RPA.

IV. Conclusion

In this paper we have presented some results obtained by using COMETA Grid infrastructures. In Section II we have shown some results obtained by using SLEPC library, a free software library for the solution of large sparse eigenvalue problems on parallel computers. Particular attention has been directed to the parallel efficiency and good results have been obtained. The use of many CPUs allows in fact to obtain a significant reduction of the consumed time and, at the same time, by distributing the data of the problem across the available processors, to perform calculations otherwise not affordable by using only one CPU.

In Section III we presented the analysis of electronic properties of Metal Clusters within the Random Phase Approximation (RPA) and two its extensions. We started, in Section III-A, by applying an improved RPA approach in order to take into account correlations in a self-consistent way, overcoming the QBA, a well known limit of RPA. In particular we have analyzed dipole plasmon modes and we have found a good agreement with experimental results, better than standard RPA. In Section III-B we have applied the SRPA to the study of collective excitations in metal clusters. We have found strong modifications of the RPA excitation spectrum, the energy being pushed

down. When the coupling of $2p-2h$ configurations among themselves is included, a further lowering is obtained even for those collective excitations whose description within RPA is widely assumed as satisfactory. Such strong modifications could be traced back to the use of QBA in calculating the expectation values appearing in the equations of motion of SRPA. Although the same approximation is used in RPA, it seems more severe in SRPA. As found in [2] within a Lipkin model, a better treatment of correlations in the ground state leads to important corrections in the energy spectrum. The results suggest that a better treatment of ground state correlations along the path of extended RPA and SRPA approaches, as done for example in [2]–[4], is necessary. Work in this direction is in progress. It is also interesting to analyze how much such strong modifications with respect to RPA description depend on the studied system. With this respect, in the near future, we plan to apply SRPA in nuclear systems by using effective Skyrme-type interactions. All these calculations require a big numerical effort: the use of Grid computing facilities will be of paramount importance.

ACKNOWLEDGMENT

This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Programma Operativo Nazionale Ricerca Scientifica, Sviluppo Tecnologico Alta Formazione (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

REFERENCES

[1] A Scalable and Flexible Toolkit for the Solution of Eigenvalue Problems, Vicente Hernandez and Jose E. Roman and Vicente Vidal, ACM Transactions on Mathematical Software, 31, 3, 351 362,2005 http://www.grycap.upv.es/slepc

[2] D. Gambacurta, M. Grasso, F. Catara and M. Sambataro, Phys. Rev. **C73**, 024319 (2006).

[3] D. Gambacurta and F. Catara Phys. Rev. **B79**, 085403 (2009)

[4] D. Gambacurta and F. Catara, Phys. Rev. **B77**, 205434 (2008).

[5] W. Ekardt, Phys. Rev. **B31**, 6360 (1985); W. Ekardt, *ibid.* **32**, 1961 (1985); M. Brack, Rev. Mod. Phys. **65**, 677 (1993).

[6] A. Bohr and B.R. Mottelson, *Nuclear Structure*, vol II, Benjamin, New York, 1975; M.N. Harakeh and A. van der Woude, *Giant Resonances*, Clarendon, Oxford, (2001).

[7] T. Reiners, C. Ellert, M Schmidt and H. Haberland, Phys. Rev. Lett. **74** 1558 (1995).

[8] J. Borggreen et al., Phys. Rev. **B48**, 17507 (1993).

A Grid-Based Comparison of Methods for the Detection of Earth-like Planetary Transits in the Time Series of CoRoT and Kepler

A. F. Lanza[1], U. Becciani[1], A. S. Bonomo[1,2], A. Costa[1], A. Grillo[3]
[1]*INAF-Osservatorio Astrofisico di Catania, Via S. Sofia 78, 95123 Catania, Italy*
[2]*Laboratoire d'Astrophysique de Marseille (UMR 6110), 38 rue F. Joliot-Curie, 13388 Marseille Cedex 13, France*
[3]*Consorzio COMETA, Catania, Italy*

***Abstract*—The space missions CoRoT and Kepler are going to observe between 60,000 and 100,000 late-type main-sequence stars to detect extrasolar planets by means of the transit technique. In other words, they will look for the small periodic light dips produced by the passage of a planet in front of its host star. Planets having a size comparable to that of the Earth p roduce dips of the order of 100 parts per million passing in front of a Sun-like star, which implies that the intrinsic stellar microvariability must be accounted for to extract the transit signal. We have developed and tested several fitting and filtering methods to find transits by terrestrial planets in the time series of late-type stars observed by CoRoT and Kepler. Specifically, large Monte Carlo experiments have been performed on the computational grid of the COMETA consortium to find out the best data processing technique to detect transits in the presence of photon shot noise and stellar microvariability for stars of different magnitudes and different parameters of the planet.**

***Index Terms*—Grid computing, astrophysics, search for extrasolar planets, astronomical photometry**

I. Introduction

THE search for extrasolar planets is one of the most interesting and active fields of modern astronomy with more than 340 giant planets discovered to date. They are mostly Jupiter-mass objects, 20 per cent of which are orbiting at distances closer than 0.1 AU from their stars. The next frontier is the search for Earth-sized objects which is possible by means of space-borne telescopes such as those of the missions CoRoT (launched on 27 December 2006) and Kepler (scheduled in March 2009). They will detect planets in an indirect way, by looking for the light dips produced by the transits of the planets themselves in front of their parent stars (see Fig. 1). The probability of a transit is R/a where R is the radius of the star and a the semi-major axis of the orbit assumed to be circular. Since $R/a < 0.003 - 0.05$ and the transit duration is of the order of a few hours or tens of hours for a solar-like star accompanied by a planet with an orbital period between 10 and 360 days, a large number of stars must be monitored for transits. CoRoT will observe about 60,000 stars, while Kepler aims at 100,000 for time intervals ranging from 150 days to 4 years. The light dips produced by the transits of an Earth-sized planet in front of a solar-like star are smaller than 100 parts per million (ppm), so that the intrinsic stellar variability must be accounted for at a 30-50 ppm level in order to be able to detect its transits. Since the transit of a large sunspot group across

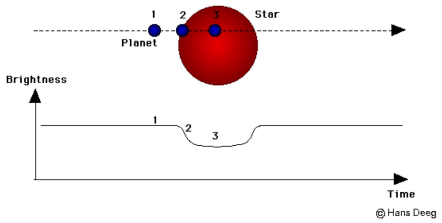


Fig. 1. Schematic view of the transit of a planet in front of its parent star, illustrating the decrease of the measured stellar flux during the passage of the planet.

the disk of the Sun produces a dip of about 3000 ppm, it is necessary to develop techniques to model the effects of stellar magnetic activity and use them to remove stellar microvariability before performing any successful search for transits by Earth-like planets.

II. Detecting transits in the presence of magnetic activity

Several techniques have been recently proposed to reduce the impact of stellar variability due to magnetic activity (i.e., cool spots, bright faculae and network) on the detection of planetary transits across solar-like stars, see, e.g., Defäy et al. (2001), Jenkins (2002), Carpano et al. (2003), Lanza et al. (2003), Aigrain & Irwin (2004), Bonomo & Lanza (2008). They are based on a variety of methods that make use of filtering in Fourier space or in the time domain, wavelet decomposition or physical models of the surface brightness inhomogeneities in order to account for the effects of magnetic activity on stellar light curves. In Fig. 2, we show the application of the fitting method developed by Lanza et al. (2003) to the transits by a planet of 2.2 Earth radii across the disk of a Sun-like star, the variability of which has been assumed to be identical to the Total Solar Irradiance time series as observed by the experiment VIRGO on board of the satellite SoHO (Frohlich & Lean 2004). The transit is not detectable by eye during phases of intermediate and maximum activity, owing to the larger intrinsic stellar

Fig. 2. Upper panel (a): the time series of the Total Solar Irradiance with superposed a planetary transit of an Earth-like planet of radius 2.2 times the radius of the Earth and an orbital period of 30 days. Lower panel (b): the residuals obtained after subtracting the model for stellar microvariability by Lanza et al. (2003).

variability. It becomes immediately visible after subtracting the best fit to the stellar variability obtained with the model by Lanza et al. (2003). Different methods have different performance according to the orbital period of the planet, the ratio between the transit depth and the standard deviation of the photon shot noise and the activity level of the star. A current area of research is the comparison of methods based on physical models of stellar variability, such as that by Lanza et al. (2003), with those based on linear or non-linear filtering or fitting techniques (see, e.g., Aigrain & Irwin 2004; Moutou et al. 2005).

III. Testing detection methods through simulated light curves

We have recently designed a large Monte Carlo numerical experiment to compare the performance of the methods proposed by Lanza et al. (2003) and Aigrain & Irwin (2004) with the best of the method proposed by Moutou et al. (2005) for the analysis of the light curves that are coming from the space experiment CoRoT. The Moutou et al's best method was

developed at Geneva Observatory and is based on the fitting of the stellar intrinsic variability with 200 harmonic components. We simulated 32,000 light curves for a Sun-like star, using TSI as a proxy for its optical intrinsic variability, spanning a large set of planetary parameters (radius, orbital period and epoch of the first transit) with one hundred different realizations of the random photon shot noise for each set of parameters. Each light curve was then fitted with both the methods of Lanza et al. (2003), Aigrain & Irwin (2004), and of Geneva Observatory's team, respectively, before performing a search for transits by means of the algorithm proposed by Kovacs et al. (2002). The large computational load of our experiment is managed by running our analyses on the grid infrastructure of the project PI2S2, allowing us to use up to 2000 CPUs in parallel (e.g., Becciani 2007). The CPU time for fitting stellar microvariability with all those methods is on average about 20 minutes, while about 10 days of elapsed time have been necessary to analyse the complete set of 32,000 light curves.

IV. Results

We show in Fig. 3 a sample of our results for the case of a planet of 1.5 Earth radii with orbital periods of 5, 10, 25 days, as labelled, respectively. The noise value (standard deviation) corresponds to a star of magnitude V=13 as will be observed by CoRoT with an integration time of one hour. The plots show the frequency distribution of the detection parameter alpha which measures the depth of a putative transit in a light curve with respect to the standard deviation of the noise (see Kovacs et al. 2002 for detail). In the present case, 400 light curves with transits have been analysed for each value of the parameters. The vertically dot-dashed lines indicate the threshold for alpha, i.e., the minimum level above which a transit is detected in the light curve with a false-alarm probability lower than 0.01. The solid blue histogram shows the fraction of detections obtained with the method by Lanza et al., while the red histogram

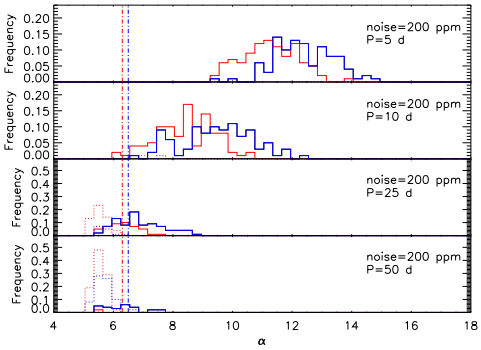


Fig. 3. A sample histogram showing the results of the analysis of a subset of our simulated light curves with the method of Lanza et al. (blue histograms) and that of Geneva Observatory (red histograms; see the text for explanation).

shows the detection with the Geneva Observatory's method. Dotted histograms show the distribution frequency of missed detections. Note that missed detections increases with increasing orbital period because the number of transits in the light curve becomes smaller giving a lower signal-to-noise ratio. Missed detection are concentrated below the threshold. Note that the method by Lanza et al. performs significantly better than the method by Geneva Observatorys team, giving systematically higher values of alpha. Such a conclusion is confirmed by the analysis of the complete set of our simulated light curves. Specifically, we find that the method of Lanza et al. is always better than that of Geneva Observatorys team when the standard deviation of the photon shot noise is at least 2-4 times larger than the depth of the transit, that is the case for stars with magnitude fainter than 13.0 in the fields of CoRoT (see Bonomo & Lanza 2008 for details). On the other hand, the Iterative Non-Linear Filter (INL) proposed by Aigrain and Irwin (2004) has a performance superior to that of the method by Lanza et al. (see Fig. 4) when the extension of the window of its sliding boxcar filter is properly selected, i.e., taking into account the level of magnetic activity of the star (see Bonomo et al. 2009, for details).

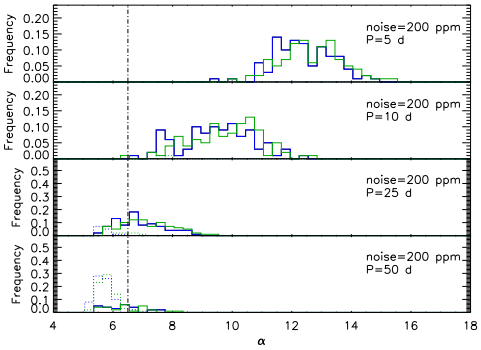


Fig. 4. As Fig. 3, but for the comparison between the method of Lanza et al. (blue histograms) and the INL filter of Aigrain & Irwin (green histograms); see the text for details on the line style coding.

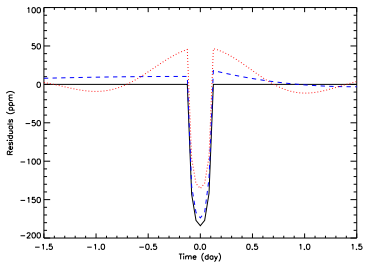


Fig. 5. The shape of a transit by an Earth-like planet as it appears in the ideal case (solid black line), in the residuals obtained with the method of Lanza et al. (dashed blue line) and in those of the method of Geneva Observatory's team (dotted red line), respectively. Note the reduction of the depth of the transit and the overshooting at the edges of the transit dip in the case of the Geneva method due to the Gibbs phenomenon.

V. Discussion and conclusions

Bonomo & Lanza (2008) shows that the advantage of the technique by Lanza et al. (2003) depends on the limited number of fitting functions used to account for stellar variability by means of a suitable physical model in comparison to the large number of harmonic components required by Geneva Observatory's method. The latter is therefore significantly affected by the Gibbs phenomenon that reduces the depth of the transits in the filtered light curves thus lowering the efficiency of detection in the presence of noise (see Fig. 5). The INL filter by Aigrain & Irwin (2004) is not affected by the Gibbs phenomenon and has a performance that can be better than that of both the other methods. Therefore, it can be the method of choice, provided that its boxcar window extension is properly fixed; otherwise, the method by Lanza et al. (2003) is recommended (Bonomo et al. 2009).

Acknowledgment

The authors gratefully acknowledge the system managers of the Trigrid and Cometa Consortia for their technical advice and kind assistance during the implementation and running of their numerical experiments on the PI2S2 grid-based high-performance computing system. ASB and AFL are also grateful to Drs. S. Aigrain, R. Alonso, P. Barge, P. Bordé, R. Cautain, M. Deluil, A. Leger, C. Moutou, for useful discussions on several aspects of the present work. The availability of unpublished data of the VIRGO Experiment on the cooperative ESA/NASA Mission SoHO from the VIRGO Team through PMOD/WRC, Davos, Switzerland, is gratefully acknowledged. The authors gratefully acknowledge support from the Italian Space Agency (ASI) under contract ASI/INAF I/015/07/0, work package 3170. This work has made use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it. Active star research and exoplanetary studies at INAF-Catania Astrophysical Observatory and the Department of Physics and Astronomy of Catania University is funded by MUR (*Ministero dell'Istruzione, Università*

e Ricerca), and by Regione Siciliana, whose financial support is gratefully acknowledged. This research has made use of the ADS-CDS databases, operated at the CDS, Strasbourg, France.

REFERENCES

[1] Aigrain, S., Irwin, M., 2004, Mon. Not. R. Astron. Soc, 350, 331
[2] Becciani, U., 2007, in Grid-enabled Astrophysics, L. Benacchio and F. Pasian, Eds., Edizioni Polimetrica; p. 179
[3] Bonomo, A. S., Lanza, A. F., 2008, Astron. Astrophys. 482, 341
[4] Bonomo, A. S., Aigrain S., Bordé P., Lanza A. F., 2009, Astron. Astrophys, in press (arXiv: 0812.1667)
[5] Carpano, S., Aigrain, S., Favata, F., 2003, Astron. Astrophys. 401, 743
[6] Defäy, C., Deleuil, M., Barge, P., 2001, Astron. Astrophys. 365, 330
[7] Frohlich C., Lean J., 2004, Astron. Astrophys. Rev, 12, 273
[8] Jenkins, J. 2002, Astrophys. J. 575, 493
[9] Kovacs, G., Zucker S., Mazeh, T., 2002, Astron. Astrophys. 391, 369
[10] Lanza, A. F., Rodonò, M., Pagano, I., Barge P., Llebaria, A. 2003, Astron. Astrophys. 403, 1135
[11] Moutou, C., Pont, F., Barge, P., et al. 2005, Astron. Astrophys. 437, 355

VisIVO: Data Visualization on the Grid

Costa A.[1], Becciani U.[1],
Antonuccio V.[1], Caniglia G.[1], Comparato M.[4], Gheller C.[2], Grillo A. [1], Krokos M.[3],
Massimino P.[1], Vitello F.[5]
[1]*INAF Astrophysical Observatory, Catania, Italy, alessandro.costa@oact.inaf.it*
[2]*CINECA, Bologna, Italy, c.gheller@cineca.it*
[3]*University of Portsmouth, UK, mel.krokos@port.ac.uk*
[4]*Dipartimento di Fisica e Astronomia, University of Catania, marco.comparato@oact.inaf.it*
[5]*CONSORZIO COMETA, fabio.vitello@oact.inaf.it*

***Abstract*—VisIVO Server is a new tool specifically designed for a grid environment. This tool offers a framework for the exploration of large-scale scientific datasets. It has been developed as a stand-alone application and a grid user can easily exploit its functionalities by means of its easy-to-use command line interface. The set of services offered by VisIVO Server include high-performance, multidimensional visualization and the huge dataset support with no fixed limit on data dimensions. The user can obtain a 3D visualization of a numerical simulation outcome with a full control of the camera position within the dataset obtaining a meaningful set of images. VisIVO Server can also offer a set of services such as randomizers, subsamplers and mathematical filters designed to facilitate a preliminary data analysis of the data being generated on the grid. We installed this software in the PI2S2 grid and, in order to facilitate the installation of VisIVO Server in any kind of grid infrastructure. We release the RPM binary package and the original source code.**

***Index Terms*—COMETA, Cosmological Simulation, Visualization, IVOA, VTK.**

I. Introduction

In many fields of science, numerical simulations represent a valuable tool for verifying theories, analysing the development and evolution of physical processes and for testing models. Present day computing systems represent virtual laboratories that can be used for virtual experiments. In some cases, e.g. in astrophysics applications, simulations are the only way to conduct experiments. The result of a numerical simulation, is an extremely large dataset: the so-called output data.

The most useful and interesting methodology for a first analysis of the output data is undoubtedly a 3D visualization. Much information, in fact, may already be obtained through a visual inspection of the data. Moreover, the visualization heightens and fosters the science opening the door to further analysis with computational tools. The tools described in this article (hereafter VisIVO Server) can be used via command line and they are designed according to the interoperability standards, set by IVOA [International Virtual Observatory Alliance], supporting various formats such as: Binary ASCII, CSV, FITS, hdf5, GADGET2 and VoTable.

II. VisIVO Server

VisIVO Server is a software package released as a suite of tools aimed at creating customized three dimensional views of cosmological data tables. These tools have been developed in c++; we have choosen this language in order to optimize the performance in terms of computational time for the data elaboration and the graphical rendering phase; we also

make a large use of external c++ libraries such as vtk [Visualization Toolkit] xerces-c++ [Xerces-C++] curl [curl]. VisIVO Server is supported for all the most popular linux platforms. Since the software design phase, we imposed therefore the support for large-scale datasets as requirement. Our product can handle any cosmological dataset without size restrictions or fixed limits regarding the dimensionality of data tables to be processed.

VisIVO Server consists of three core components: **VisIVO Importer**, **VisIVO Filter** and **VisIVO Viewer**. We have structured VisIVO Server components so that the user can accomplish a complex task by organizing a pipeline of commands. To create customized views of 3D renderings from astrophysical data tables, a two-stage process is employed. First, VisIVO Importer is used to convert user datasets into VisIVO Binary Tables. Then, VisIVO Viewer is used to display customized views of 3D renderings. As an example, to show a view representing three columns of an astrophysical data table supplied in ascii form, say col_1, col_2 and col_3, the following commands are employed:

- **VisIVOImporter** –fformat ascii UserDataSet.txt
- **VisIVOViewer** -x col_1 -y col_2 -z col_3 –scale –glyphs pixel VBT.bin

VisIVO Filter is a suite of filters used to generate new VisIVO Binary Tables from existing ones, e.g. consider constructing a new version of a user dataset through randomized sub-sampling in order to fit it in the available RAM.

A. VisIVO Binary Table

A VisIVO Binary Table (hereafter VBT) is an internal binary representation. We introduced a new data representation in order to improve the efficiency and the modularity of our product. VBT is composed of a header file (extension .bin.head) containing all necessary metadata, and a raw data file (extension .bin) storing actual data values. The header will contain information regarding the overall number of fields and the number of points for each field (for a point dataset) or the number of cells and relevant mesh sizes (for a volume dataset). The so-called raw data file (extension .bin) is a data file composed of the sequence of values representing each field, e.g. all X followed by all Y values.

The header file allows the user to describe a point dataset as well as a volume dataset. It contains information about the data as regards endianism (i.e. little/big), data type (i.e. float/long), number of field and their name. For a volume dataset it also contains the mesh geometry for each coordinate and the size of volumetric cells. The header file contains the following fields:

```
f | d
n1
n2 [X Y Z DX DY DZ]
l | b
label1
label2
label3
...
label(n1)
```

Where

- **f,d** is the data type: f=float d=double;
- **n1** denotes the number of columns (fields) in the VBT;
- **n2** denotes the number of elements of each field;
- **[X Y Z]** (optional) represent the number of cells in each dimension of the mesh size (used only for Volumes);
- **[ΔX ΔY ΔZ]** (optional) represent the real unit of each cell (used only for Volumes);
- **l,b** denotes the endianism (little/big);
- **label**$_1$**....label**$_{n1}$ are the field names.

B. VisIVO Importer

VisIVO Importer is the tool that converts input data into the internal binary (VBT) used by VisIVO Filters for data processing and by VisIVO Viewer for display. The conversion is optimized and is independent of the data size. More often a researcher deals with files hosted

by remote archives and reachable through protocols such as FTP, HTTP, SFTP. In order to meet this requirement we have improved VisIVO Server functionalities with a full control of such protocols. VisIVO Importer is therefore able to handle both local and remote input files. The user can employ a URL (Uniform Resource Locator) syntax in order to identify a remote input file to be passed to VisIVO Server and the file is automatically handled by the Importer. A login and password information can be optionally passed to the command. VisIVO Importer has the support to convert the following data format:

- **ASCII** and **CSV**
  ASCII files are expected to be in tabular form. An ASCII file may contain values for N variables organized in columns. The columns are typically separated by a white space character, e.g. spaces or tabs. The first row of an ASCII file lists the N variables names explicitly. CSV (Comma Separated Values) has fields (columns) separated by the comma character and records (rows) separated by newlines.
- **VOTable**
  The VOTable format is an XML standard for the interchange of data represented as a set of tables. A table is an unordered set of rows, each having a uniform format, as specified in the table metadata information. Each row in a table is a sequence of table cells, and each of these contains either a primitive data type or an array of such primitives.
- **FLY**
  It is a code that uses the tree N-body method, for three-dimensional self-gravitating collisionless systems evolution. FLY [Becciani et al. 2003] output format is a binary sequence of values of n data points as follows: X1,Y1,Z1, X2,Y2,Z2 ,.... Xn,Yn,Zn, Vx1,Vy1,Vz1, Vx2,Vy2,Vz2, ... Vxn,Vyn,Vzn.
- **FITS Table**
  The definition of FITS is a codification into a formal standard, by the NASA/Science Office of Standards and Technology (NOST), of the FITS rules endorsed by the IAU FITS Working Group ([IAU-FWG]). FITS supports tabular data with named columns and multidimensional rows.
- **GADGET**
  It is freely available code for cosmological N-body/SPH simulations on massively parallel computers with distributed memory. GADGET [Springel et al. 2001] uses an explicit communication model that is implemented with the standardized MPI communication interface.
- **Raw Binary**
  Raw files are simply a binary dump of the memory for data points. The content of the Raw Binary data points file is a sequence of x, y and z coordinates for each point followed by a sequence of scalar fields.
- **Raw Grid**
  Raw files are simply a binary dump of the memory. The content of a volume file is a sequence of values: one value for each mesh point.
- **TVO XML**
  This document is a VOTable that must contain all the key informations to download data. It is used to describe the content of a generic binary file, and it contains the reference to input files. Each row in the document will be a field of the output file. All the rows with the same species identifier are included in the same file. If a filename starts with http://, sftp:// or ftp:// the remote file is downloaded using the CURL library. The TVO XML document is under revision by the TVO Interest group in the IVOA framework.

C. VisIVO Filter

VisIVO Filter is a collection of data processing modules that can process one or more VBTs producing a VBT. The filters support a range of operations such as: scalar distribution, mathematical operations, selections of regions, decimation and so on. The selection

and randomization operations are particularly important as they are typically employed for the construction of reduced data wich can be used directly by VisIVO Viewer or the VisIVO Desktop application. The general Syntax is
$VisIVOFilters\ -\ -opfilterOpCode\ <\ options > [--file]InputFile$
where
the *filterOpCode* selects a specific operation to be processed (i.e. randomization), options are related to the specific selected operation and the *InputFile* is an input file or a list of files. In the following we describe some outstanding filters.

Randomizer
This operation creates a new VBT sub-table as a random subsample from the input table. The operation allows the user to select a percentage of the original table in order to proceed faster in the data analysis or to visualize data using VisIVO Viewer. An optional seed can be given to initialize the pseudorandom number generator. Values are extracted in a random sequence. A similar VisIVO Filter operation (*decimator*) allows you to select a subset of data but in a fixed sequence.

Table Merger
This operation creates a new VBT from two or more VBTs. For each input table the Merge operation can use all or a subset of a VBT column. In case of a VBT having not all columns with the same dimension, the operation creates a new VBT having the size of the smallest or of the largest table. In the first case longer columns will be truncated in the second case smaller columns are padded with a given value.

Subregion Extractor
This operation creates a new VBT from an input one as a sub-box or a sub-sphere region. There are two different geometries that can be selected:

- *Sphere*. The center and the radius of the sphere are used to identify the sub region to be extracted.
- *Box*. The length of the box side and the corner position (or the box center position) are used to fix the sub region.

This operation can be used to investigate only a subvolume from the original one. All the columns of the input VBT are reported on the output VBT, and the analysis on all the fields of a given dataset could be done only on a specific region but with a reduced dataset.

Select Rows
This operation creates a new VBT setting limits on one or more fields of the input VBT. The operation must have a list of columns names, and the limits (max-min) where data will be considered. Limits on more than one column can be combined with OR or AND operators. The rows that satisfy the given limits are reported on the output VBT. The output VBT contains all the columns of the input VBT. Even in this case we can say that the analysis on all the fields of a given dataset could be reduced only to a specific region of the multiphase space.
This operation cannot be applied to volume data.

Mathematical Operations
This operation lets the user derive a new field starting from the original ones. It adds a new column to the input VBT or optionally creates a new VBT using mathematical operators.

Point Distribute
This operation transforms a VBT representing points in a VBT wich represents a volume by distributing a scalar value or a constant value on to a regularly spaced cubic mesh with a user defined resolution.

D. VisIVO Viewer

VisIVO Viewer (fig. 1, fig 2) is based on the Visualization ToolKit version 4.0 ([Visualization Toolkit]) library for multidimensional visualization. It creates 3D png images of a dataset, it can handle both data points and volumes. The visualization process can be given on a large but limited number of elements. The images are created by loading all input data on the available RAM. The visualization process of huge data relies on the facilities provided by

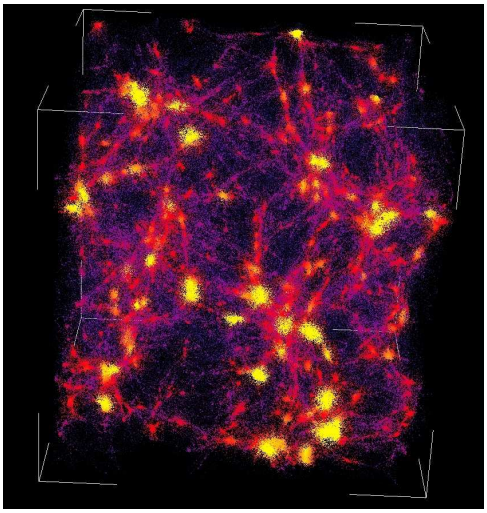


Fig. 1. A VisIVO Server image showing the whole computational box of a Λcdm cosmological simulation with 512 Million of particles

the other tools of VisIVO Server. In particular we can suggest the following pipeline:

- VisIVOFilter runs the *Visual* operation (a specialized *Randomizer* operation) on large data. This operation reduces the original dataset to fit the available memory and produces a new VBT for the visualization.
- VisIVOViewer processes the VBT given by the above step.

VisIVO Viewer can render points, volumes and isosurfaces within a bounding box used for representing the coordinate system employed. Moreover there is support for customized look up tables and for the use of a variety of glyphs, such as cubes, spheres or cones. The standard output of VisIVO Viewer consists of four images corresponding to fixed camera positions and zoom factors and another image corresponding to user-defined line command options.

III. VisIVO: Grid use case

The VisIVO Server is optimized for running on Scientific Linux $x86-64$ platform with

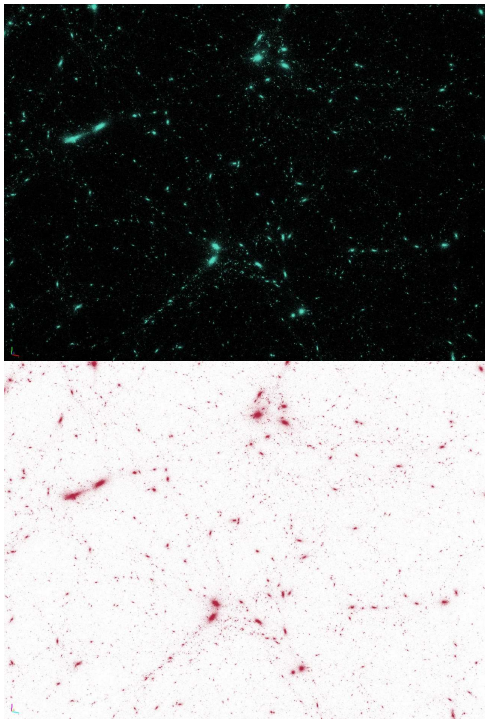


Fig. 2. Two VisIVO Server images showing some details in a suggestive view of a Λcdm simulation with 512 Million of particles. The second image is a color inversion.

GLite 3.1 GRID Middleware. The installation of this package is quick and easy, and it is performed with an RPM file. This file contains the three executables and two shell scripts for the automatic configuration of the environment path for both bash and tcsh shells. The use of VisIVO Server in a Grid environment is quite simple, you have to create a jdl file like this:

```
Type = "Job";
JobType = "Normal";
Executable = "Start.sh";
StdOutput = "Std.out";
StdError = "Std.err";
InputSandbox = {"Start.sh"};
OutputSandbox = {Std.out",
      "Std.err"};
RetryCount = 0;
```

Because of the high size of the input files, it is recommended to use the Catalog Data Management on the Storage Element. This file launches the executable bash script Start.sh; here is an example:

```
#!/bin/sh

export LOCAL_DIR=$(pwd)
export LFC_HOST=lfc-01.ct.\
 trigrid.it
echo ""

lcg-cp --vo cometa \
 lfn://grid/cometa/visivo/ \
 cluster.ascii \
 file://$LOCAL_DIR/cluster\
 .ascii \

VisIVOImporter --fformat ascii \
 --out cluster cluster.ascii

VisIVOFilters --op randomizer \
 --file cluster.bin --perc 0.8 \
  --out clusterRand.bin

VisIVOViewer --out vsimage \
 --opacity 0.8 clusterRand.bin

ls -l

for img in $(ls *.png)
do
   lcg-cr --vo cometa -l \
     lfn://grid/cometa/test/ \
     $img file://$LOCAL_DIR/ \
     $img -d \
     inaf-se-01.ct.pi2s2.it
done
```

In this example the lcg-cp command downloads the $cluster.ascii$ file from the Storage Element through the Grid Catalog, then we launch the VisIVO Server that consists in 3 steps:

- VisIVOImporter takes the $cluster.ascii$ file as input and returns 2 files: $cluster.bin$ and $cluster.bin.head$;
- VisIVOFilters takes the $cluster.bin$ file as input and extracts a randomized subset of it in the file $clusterRand.bin$;
- VisIVOViewer takes the $clusterRand.bin$ file as input and produces 4 different images in PNG format.

Finally the lcg-cr command stores the 4 images in the Storage Element.

IV. Conclusion

The VisIVO Server software is distributed under GPL license (www.gnu.org) for non-commercial use only. It is an open source project and it can be directly downloaded from the sourceforge code repository [VisIVOServer at Sourceforge]; users and researchers can join the developer team to enhance and extend the tool functionalities.

Acknowledgment

This research has made use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministero dell'Istruzione, Universita' e Ricerca (MIUR) within the Piano Operativo Nazionale Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

We wish to thank Mrs. Luigia Santagati for the revision of the English text.

References

[International Virtual Observatory Alliance] http://www.ivoa.org

[Visualization Toolkit] http://www.vtk.org

[IAU-FWG] http://fits.gsfc.nasa.gov/iaufwg/iaufwg.html

[Xerces-C++] http://xerces.apache.org/xerces-c/

[curl] http://curl.haxx.se/

[Becciani et al. 2003] U. Becciani, et al., *FLY. A parallel tree N-body code for cosmological simulations* 2003, CoPhC. Vol. 155, Issue 2, p. 159-179

[Springel et al. 2001] V. Springel, et al., *GADGET: a code for collisionless and gasdynamical cosmological simulations* 2003, CoPhC. Vol. 155, Issue 2, p. 159-179

[VisIVOServer at Sourceforge] https://sourceforge.net/projects/visivoserver/

A Web Application for Photometric Redshift Estimation

O. Laurino[1], R. D'Abrusco[1,4], M. Brescia[2], S. Cavuoti[1], A. Corazza[1], G. d'Angelo[1], C. Donalek[3], S.G. Djorgovski[3], N. Deniskina[1], M. Fiore[1], M. Garofalo[1], G. Longo[1,2], A. Mahabal[3], F. Manna[1], A. Nocella[1], B. Skordovski[1]

[1]*Department of Physical Sciences, University Federico II, Naples, Italy*

[2]*INAF-OACN, National Institute of Astrophysics, Naples, Italy*

[3]*CALTECH, California Institute of Technology, Pasadena, California USA*

[4]*Department of Astronomy, University of Padua, Padua, Italy*

laurino@na.infn.it

***Abstract*—In the era of massive astronomical datasets, efficient identification of candidate quasars and the reconstruction of their three dimensional distribution in the Universe is a key requirement for constraining some of the main issues regarding the formation and evolution of QSOs. A method for the determination of photometric redshifts of QSOs based on multiwavelength photometry and on a combination of data mining techniques will be discussed. This procedure, specifically suited for accompanying the candidate selection method discussed in (D'Abrusco et al. 2008), makes use of specific tools developed under the EuroVO and NVO frameworks for data gathering, pre-processing and mining, while relying on the scaling capabilities of the computing grid. This method allowed us to obtain photometric redshifts with an increased accuracy (up to 30%) with respect to the literature.**

***Index Terms*—QSO, AGN, neural networks, photometric redshifts.**

I. Introduction

THE accurate knowledge of the shape and redshift evolution of the quasar luminosity function (QLF) is fundamental to many fields of modern cosmology [1]. For instance, the faint-end slope of the QLF, which is found to decrease with redshift (e.g. [2], [3], [4]), is important in determining the early formation history of Black Holes (BHs) and their contribution to reionization, as well as the possible connections between quasars and, e.g., the low-luminosity Seyfert galaxies seen at $z \simeq 0$. In traditional models, this trend requires a significant and rapid evolution in the shape of the distribution of the masses of the host galaxies, which cannot be accounted for in either semianalytical models or numerical simulations and is not consistent with a wide range of galaxy observations (e.g., [5]). These models, in fact, generally succeed at high redshift but do not explain the decrease in counts of bright quasars at low redshift (e.g., [6]), unless other ad hoc mechanisms are invoked such as, for instance, the suppression of the growth of high-mass spheroids (e.g., [7]), or the evolution in the BH accretion efficiency with redshift (e.g., [8]). In other words, these models cannot be extrapolated to low luminosities or to redshifts where the slope is undetermined. Observations at high redshifts are uncertain, and will remain so until large, uniformly selected samples will allow to measure the faint-end slope at both low ($z < 1$) and high ($z > 3$) redshifts. The construction of such samples is troublesome due to the necessity of obtaining spectra for large samples of mainly faint objects: this task which is both very demanding in terms of observing time and challenging in terms of signal to noise ratio. In recent years, however, the availability of deep, accurate and multiband digital surveys such as the Sloan Digital Sky Survey (SDSS, [15]), the

Two Degree Fields (2dF), or the UKIDSS ([16]) has opened a new possibility: the search for quasars in the photometric parameter space and the evaluation of their distances using photometric redshifts. This approach uses photometry to determine the redshift of the observed object relying upon the fact that the spectrum of radiation emitted by the sources possess very strong features that are detected in different filters of the photometric system. The photometric redshift of a source can then be calculated by establishing an empirical relation between the brightness in different filters and the actual redshift of the source, and finally applying such relations to sources for which only the photometric parameters have been measured.

Such approach, however, is constrained by the availability of a reliable (in terms of completeness and efficiency) sample of candidate quasars, and the accuracy with which it is possible to obtain an estimate of their distance. As it has been shown in several papers (cf. [9], [11]), the first task has already been successfully completed but the second step still poses few problems.

In what follows we shortly outline a new method based on Neural Networks which, by making use of the S.Co.P.E. computing infrastructure, offers to the Virtual Observatory (VObs) community the possibility to derive photometric redshifts for both galaxies and quasars, provided that a large enough spectroscopic base of knowledge on which to train the algorithms is available. The application of machine-learning techniques for the evaluation of photometric redshifts of galaxies has been widely explored in the literature (cf. [13] and references therein) but the problems posed by quasars are very different. In the case of these type of extragalactic sources, in fact, the base of knowledge (BoK) is characterized by a high degree of sparseness in the parameter space since quasars are observed in a very large interval of redshifts (from ~ 0.5 to ~ 6) and cover a large region of the photometric space so that the average density of BoK members can be very low. The application of this method to the catalogue of candidate quasars produced by [11] will lead to the evaluation of the luminosity function of candidate quasars in the optical magnitudes of the SDSS ([14], in prep.).

II. The Method

ONE possible approach to the estimation of the photometric redshifts is based on supervised machine learning algorithms such as the neural networks [12], trained on a subsample of the photometric data set for which spectroscopic redshifts are available (the BoK). The method is then capable of producing accurate estimates of the photometric redshifts for sources found in the same regions of the photometric parameter space spanned by the members of the BoK.

The photometric redshift estimation can be regarded as a regression problem: we seek out a functional mapping

$$f : c \in \mathcal{C} \rightarrow z_{phot} \in \mathcal{Z}$$

from the parameter space $\mathcal{C}$ of the source's colours to the target space $\mathcal{Z}$ of the redshifts.

A. The Multi-Layer Perceptron

To solve regression problems, the Multi-Layer Perceptron (MLP) is one of the most robust and reliable non-deterministic, machine learning techniques: i) it is a universal function approximator and does not require any assumptions on f; ii) it has a good response when dealing with noisy data and, iii) it has remarkable generalization properties. In a nutshell, the problem is the following: we need to *train* an MLP network to *learn* the mapping function f.

As for every supervised machine learning algorithm, it requires the BoK to be split into different subsets:

- *Training Set*. 60% of the objects will be used to provide the network with *examples* to induce the mapping function form;
- *Validation set*. We save 20% of the data to avoid overfitting: in principle (and in the absence of degeneracies), a neural network could approximate a function with arbitrary

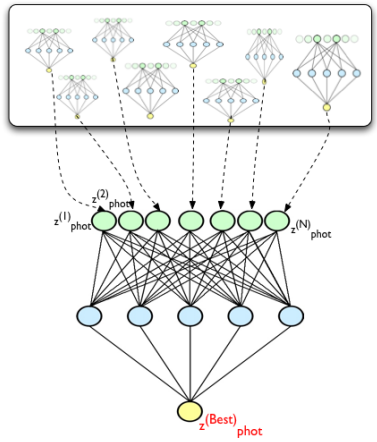


Fig. 1. Multilayer Perceptrons in ensemble

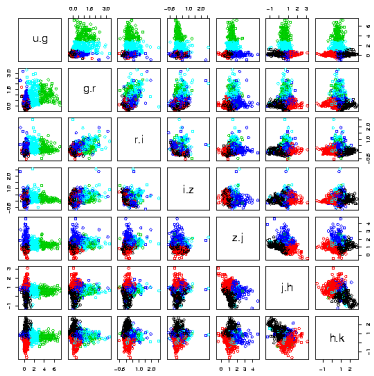


Fig. 2. Clustering of a multidimensional dataset with k-means

precision given a sufficient number of hidden nodes. We stop the training phase when the error on the validation set is minimum.

- *Test set*. The remaining objects (20%) are eventually used to test the overall performance of the network.

In the specific case of quasar redshifts, the regression problem suffers from the intrinsic deficiency of the data due to the noise induced by the *degeneracies*. So, it is preferable to train an MLP *ensemble*, where each network is trained (and optimized) on a specific partition of the parameter space. We wish to recall that MLP training is an NP-hard optimization problem, so it can be computationally expensive, with a computational cost that grows linearly with the number of input patterns and the dimensionality of the features space.

B. Partitioning The Parameter Space

Clustering is an unsupervised data mining method for organizing objects into groups according to a given definition of distance. We can also look at this data partitioning method, from a geometrical standpoint, as providing a way to divide the parameter space into different regions according to a given criterion. It has to be stressed that, as for all the unsupervised methods, there is not a unique criterion to decide whether a specific partition is better than another; neither we know the best number of clusters to partition the parameter space in. Again, we can exploit our Base of Knowledge to set a *feedback* criterion based on the final regression results to infer the best number of clusters.

k-means is one of the simplest, yet robust, clustering algorithms: given the number of clusters (k) and using an euclidean distance, k-means finds the k cluster centroids aggregating inputs with similar properties. When splitting the dataset into clusters we decimate the patterns the neural networks will be trained on. This could not be much of a problem when the parameter space is densely covered by the patterns and there is a large number of training objects. However, when dealing with QSO catalogues, the sparseness of the data does not allow to ignore the effects of this decimation on the training process, and it is necessary to find a different training strategy. This can be done by taking redundant, overlapping clusters so to introduce a coupling in the way network are trained. This coupling can be exploited with the *ensemble* approach itself, as it will be shown in the following sections.

To produce such redundant clustering we made use of a fuzzy variant of the k-means clustering algorithm: each point in the parameter

space belongs to each cluster with a non-zero membership probability; we then set an arbitrary threshold T, so that a cluster actually contains only points with a membership larger that T. With an iterative procedure the fuzzy k-means algorithm is run several times with a variable number of clusters between a minimum and a maximum value. The optimal number of clusters N is defined by the value for which the overall regression error is minimized.

C. Neural Networks ensemble

Given N different clusters, each cluster represents the BoK for a different MLP network, and these BoKs share a certain variable number of sources, with the effect of introducing a coupling in the training of the different networks. In other words, each of the N networks is an *expert* for the sub-domain of the parameter space it has been trained on, and these sub-domains are overlapping. Without the *ensemble* approach, in order to estimate the photometric redshift of a generic point (source) $c \in \mathcal{C}$, we should assign the point to one of the k clusters in the first place, and then run the correspondent MLP network. Alternatively, in our method, each input pattern c is presented to all the MLPs, allowing them all to provide their "opinion", and then these opinions (i.e. different guesses of the photometric redshift) are combined to provide our best estimation. According to the way we determined the clusters, the outputs of the single networks will be coupled and different networks will have a different reliability on different regions of the parameter space. The combination is then performed by means of a *gating* MLP network that *learns* how to combine the predictions coming from the first layer networks.

We measure the performance of the method by calculating the Mean Square Error over the test set. Each different clustering scheme will yield to a different MSE value. We define the best number of clusters N as the number that yields to the minimum value of the MSE error over the test set.

D. The method as a whole

The workflow we have briefly depicted so far, involves several optimization problems. Given the smallest and largest number of clusters we want to probe, N_m and N_M, for all k between N_m and N_M we have to:

- perform a k-means clustering to find the k centroids;
- train k different MLP networks.

The number of first layer MLPs to train is

$$\begin{aligned} N_{net} &= \sum_{k=N_m}^{N_M} k \\ &= \frac{N_M(N_M+1)}{2} - \frac{N_m(N_m-1)}{2} \end{aligned}$$

If $N_{clu} = N_M - N_m + 1$,, the total number of optimization problems is $N_p = N_{net} + 2N_{clu}$,, since we have to take into account the gating networks and the clusterings. For $N_m = 2$ and $N_M = 10$, we have 72 optimization problems. However, since the k-means's time complexity is linear in all the relevant factors (iterations, space dimensionality, number of clusters, number of points), we can neglect the computational cost of the clustering task and consider just the 63 MLP training tasks, i.e. 63 NP-hard optimization problems. All these tasks are obviously independent, so they can be launched in parallel on the computing GRID.

E. The data

We performed a set of experiments aimed at the estimation of photometric redshifts for quasars candidates extracted from the optical parameter space of SDSS [15] stellar sources defined by the colours $(u-g, g-r, r-i, i-z)$. These candidates were obtained via an original method based on unsupervised clustering (cf. [11]) which has yielded a catalogue of QSO candidates in the SDSS (cf. the webpage at voneural.na.infn.it/qso.html). The colours of the candidate quasars were obtained using the *psfgMag* magnitudes retrieved from the $PhotoObjAll$ table of the SDSS DR7 database, after culling the

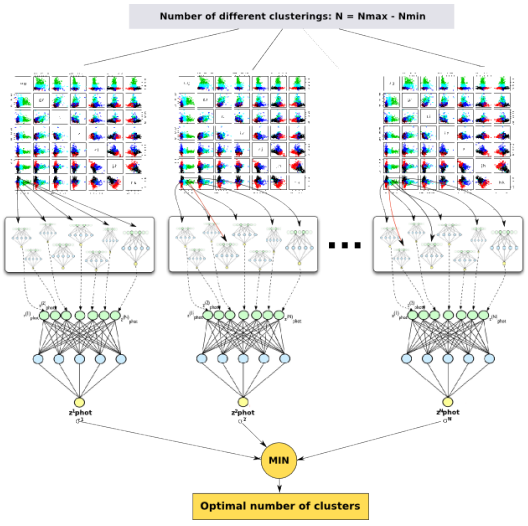


Fig. 3. A flow chart of the whole method.

dataset excluding all sources with missing magnitudes or NaN. As BoK, we used the spectroscopic redshifts obtained from the SDSS spectra for a subsample of photometric candidates down to a limiting magnitude g 17.7. In order to obtain a clean sample of confirmed quasars and to avoid the introduction of biases in the training process of the neural networks, only spectroscopic redshifts with confidence $zConf > 0.90$ were included.

F. A preliminary result

As first scientific application, we have obtained photometric redshifts for a sample of optical candidate quasars extracted from the SDSS, obtaining the accuracy (measured as the robust standard deviation of the $z_{phot} - z_{spec}$ variable for the quasars of the test set) $\sigma = 0.23$, which is better than the accuracies of most of the methods found in the literature (e.g. [9], [10]).

III. The method as a VObs Web Application

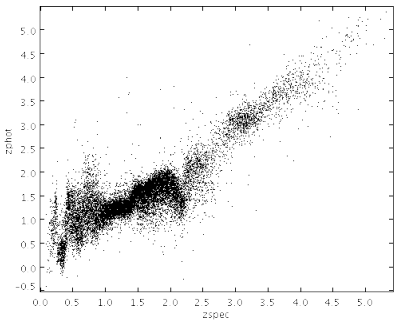


Fig. 4. z_{phot} vs z_{spec} scatter plot: $\sigma = 0.23$. The dataset is a catalogue of optical candidate quasars extracted from the SDSS. See text for details.

THE VObs aims at providing the world astronomical community with a comprehensive, consistent and interoperable infrastructure for sharing and federating the massive amounts of astronomical data observed or simulated by astronomers worldwide. By means of the VObs it has become possible to access a multidimensional parameter space and to retrieve the most complete Base of Knowledge available. For photometric redshifts this means that we can build rich multiwavelength BoKs significantly improving our prediction accuracy, at the price of increased computational cost. Supervised methods need to be trained on a well defined BoK, and their predictions are reliable as far as the input patterns fall in the same regions of the BoK they have been trained on. Photometric redshift estimation can obviously improve as new colours are added to the feature space, but in order to exploit the information carried by this extension of the parameter space, specific networks need to be trained on the new parameter space in the first place. This is the main reason why, since its start, the method presented above was conceived as a Virtual Observatory (VObs) tool to be offered to the community as a web application through the DaME/VOneural platform. DAME (Data Mining & Exploration) is described elsewhere in this volume [17]. In its final version, our application will:

- ask the user for a catalogue of extragalactic sources, defined by their names or their sky

coordinates;
- for each source, gather all the relevant information within the VObs;
- dynamically call the properly trained models;
- if the spectroscopic redshift is available for a source, retrieve it;
- output the original catalogue with some columns added: the photometric redshift, the spectroscopic (if any) redshift and the respective uncertainties;
- optionally, retrieve from the VObs some general information about the source, and display it to the user.

A prototype of this web application is already served along with the DAME/VONeural front end at the address http://dame.na.infn.it

IV. Conclusions

OUR method for photometric redshift estimation of quasars was designed to improve upon the accuracy of available methods. The key role is played by the way we handle the intrinsic deficiency of the base of knowledge: for quasars, the BoK can be very sparse, and degeneracies may likely be introduced by the spectroscopic features shifting off the photometric system filters used for the observations. To overcome this deficiency we apply, for the first time in astronomy, fuzzy clustering techniques to the original catalogue, in order to implement redundant training sets. We train an *ensemble* of artificial neural networks on these subsets, so that the networks become *expert* at specific overlapping regions of the feature space. Input patterns are then presented to all the experts and a gating neural network is trained on their outputs in order to combine them. The subsets are determined by the fuzzy k-means clustering algorithm. This method, already publicly available as a web application at the address http://dame.na.infn.it, exploits the computational capabilities of the S.Co.P.E. computing GRID and is fully compliant with the Virtual Observatory standards and infrastructure.

Acknowledgments

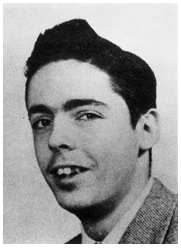
This work was funded by the S.Co.P.E. project, by the VO-Tech and VO-AIDA European projects and by the DAME project financed by the Ministry of Foreign Affairs as a "Great Relevance Bilateral Project" between Italy and U.S.A.

References

[1] Hopkins P.F., Hernquist L., Cox T.J., Robertson B., Di Matteo T. & Springel V., 2006, ApJ, 639, 700.
[2] Hunt, M. P., Steidel, C. C., Adelberger, K. L., & Shapley, A. E. 2004, ApJ, 605, 625
[3] Cirasuolo, M., Magliocchetti, M., & Celotti, A. 2005, MNRAS, 357, 1267
[4] Hasinger, G., Miyaji, T., & Schmidt, M. 2005, A&A, 441, 417
[5] Merloni, A. 2004, MNRAS, 353, 1035
[6] Enoki, M., Nagashima, M., & Gouda, N. 2003, PASJ, 55, 133
[7] Scannapieco, E., & Oh, S. P. 2004, ApJ, 608, 62
[8] Haiman, Z., & Menou, K. 2000, ApJ, 531, 42
[9] Richards, G. T. et al. 2008, ApJS, 180, 67-83.
[10] Weinstein, M. A., Richards, G. T. et al., 2004, ApJS, 155, 243
[11] R. D'Abrusco, G. Longo and N.A. Walton, accepted by MNRAS, 2008.
[12] R. D'Abrusco et al. 2007, ApJ, 632, 752.
[13] R. D'Abrusco, PhD Thesis, University of Napoli Federico II, 2007.
[14] O. Laurino et al., in preparation, 2009.
[15] C. Stoughton et al. 2002, AJ, 123, 485-548.
[16] A. Lawrence et al. 2007, MNRAS, 379, 1599-1617.
[17] Brescia M., et al. 2009, these proceedings.
[18] Bishop, C. M. 1995, Neural Networks for Pattern Recognition, New York, Oxford Univ. Press.

Omar Laurino is both a physics Master of Science student and an IT worker. He worked as a system administrator for both the Italian Institute of Nuclear Physics (INFN) and the Department of Physical Sciences of the University of Naples, and is now taking (at last) his *laurea* in physics with a thesis on the reconstruction of the three-dimensional distribution of candidate QSOs with photometric redshifts and the derivation of their luminosity function.

He is the Project Engineer of the DaME/VONeural project, a data mining framework for massive datasets (also in these proceedings).



Raffaele D’Abrusco is currently a post-doc researcher at the Department of Astronomy of the University of Padova, after having spent 1 year as post-doc at the University of Naples. He earned his PhD in Physics at the same University of Naples ”Federico II” on 21st December 2007 defending a thesis titled ”The Large Scale Structure of the Nearby Universe”. His area of scientific interests is the extragalactic astronomy, in particular the study of the large scale distribution of galaxies, the observational characterization of AGNs and QSOs and the application of new data mining tools and statistical techniques to several astronomical topics. He collaborates with the DaME/VONeural group.

Simulations of Cosmic Strings Signatures in the CMB

Riccio G.[1,2,3], D'Angelo G.[1,3], Sazhin M.V.[4], Sazhina O. S.[4], Longo G.[1,2,3] and Capaccioli M.[3]

[1]*University of Naples Federico II, Via Cinthia 6, Napoli, Italy*

[2]*INFN - Napoli Unit, Via Cinthia 6, Napoli, Italy*

[3]*INAF - Osservatorio Astronomico di Capodimonte, Salita Moiariello 16, Napoli, Italy*

[4]*Sternberg State Astronomical Institute of Moscow State University, Universitetskii pr.13, Moscow, Russian Federation*

***Abstract*—Theory shows that a moving string should produce a step-like discontinuity of low S/N ratio in the Cosmic Microwave Background (CMB) as a consequence of the Doppler shift due to the relative velocity between the string and the observer. In the simplifying assumption that the string is a straight discontinuity in space time, we used the S.Co.P.E. computational grid to produce a large number of simulations covering a wide range of values for the velocity of the string, its direction of motion and its distance from the observer. The simulations were divided in 2 types: i) without string signature, but simulating the signal and noise structure of the observed WMAP data, and ii) "without noise with string signature". The two types of data were then combined to produce realistic simulated images sampling the whole parameter space defined by S/N and string parameters. The simulated images will then be used to investigate the performances of different types of filters. The first filter divides the images into strips of $\theta = 10^{\circ}$ starting from the equator; each strip is then "squeezed" so as to reduce it to a 1-D vector containing the value of temperature averaged over θ of the strip. Successively, all squeezed strips are summed together and the resulting vectors are searched for systematic trends of temperature being over 2σ from the mean value of the distribution. Statistical analysis shows that this filter has an efficiency of about 0.75 in recognizing the presence of a string. The same efficiency is reached by the filter when tested on "only CMB" maps to evaluate the contamination from spurious events.**

***Index Terms*—cosmic string, anisotropy of cosmic microwave background radiation, cosmology.**

I. Introduction

COSMIC strings were postulated by Kibble in 1976 [1] as a special type of topological defects of the space-time and immediately became a hot issue in both theoretical physics and cosmology [2], [3]. In a nutshell, it can be stated that cosmic strings are infinitely long and filamentary remnants of primordial dark energy which formed in the early Universe, and were then stretched by the cosmic expansion up to the point that, at present epoch, some cosmic strings could cross the entire length of our observable Universe. With respect to other topological defects, cosmic strings find their natural explanation in the inflationary scenario and are fully predicted by a wide class of elementary particle models. For instance, in the superstring theory (both in the compactification models and in theories with extended additional dimensions) cosmic strings arise as a natural consequence. While all theories predict for cosmic strings (either classical strings, or F- and D-strings) an extremely long cosmological length and a negligibly small cross-section, other parameters (e.g. the linear density) strongly depend on the underlying model and can vary over a wide range [4], [5]. The detection of a cosmic string would therefore allow to disentangle the true

underlying theoretical framework and to extend our experimental knowledge in an energy domain which is unavailable to modern and foreseen particle accelerators (in the energy range of GUT, i.e. $10^{14} - 10^{16}$ GeV). In spite of their relevance, so far there are no observational evidences to prove or disprove the existence of cosmic strings. Theory shows, however, that their signatures can be found:

1) in optical surveys by looking for gravitational lensing events;
2) in radio surveys by investigating the structure of CMB anisotropy; This issue has become extremely fashionable in prevision of the launch of the Planck mission;
3) by looking for model dependent and rare features, such as gravitational radiation from string loops and from straight string, interactions of strings with black holes, decay of heavy particles emitted by the string, interaction of two strings etc.

Furthermore, several recent results based on the analysis of the WMAP 5-years data, show that cosmic strings are not a primary source of primordial density perturbations and therefore are not responsible for large scale structure formation. This implies that their density in the primordial universe must be very low and that the existence of cosmic string networks is very unlikely. These results, however, do not rule out the existence of individual, isolated strings as those which are investigated in the present paper. Theory predicts that cosmic strings may possess a wide range of parameters: with linear density varying from the scale of the electroweak unification to GUT scales, and with velocities from zero to the speed of light. Finally, the string geometry can vary from straight to loop like structures.

II. Cosmic strings and CMB

As the Universe expands, the temperature of the plasma falls down to the recombination temperature, the formation of matters becomes energetically favorable, and the photons begin to propagate freely in space. producing the so called surface of the last scattering (LSS). There is a small anisotropy in the temperature of the LSS, of the order of 10^{-5} which can be modeled as a multipole. In the low-multipole part of the angular spectrum, the Standard Cosmological Model is not in very good agreement with observational data and such disagreement can be traced to several causes among which a nontrivial topology of the Universe, or the presence of one or at most a few cosmic strings. In order to test the latter hypothesis, we investigated the signature left on the CMB by a single straight infinite cosmic string moving with constant velocity in a homogeneous and isotropic background. A moving string might generate a large scale CMB anisotropy according to a simple Doppler mechanism. In a cosmological setting, the string is backlighted by a uniform blackbody radiation background, and the Doppler shift results in a discontinuous change of the background temperature across the string[10], [11]. The magnitude of this variation is:

$$\frac{\delta T}{T} \approx 8\pi G\mu\gamma\beta,$$

where μ is the linear density of string and, as usual, $\gamma = 1/\sqrt{1 - v^2/c^2}$, and $\beta = \frac{v}{c}$ is the projection of string velocity on the line perpendicular to that one joining the source and the observer.

According to the simple model discussed in a previous paper [13], the CMB anisotropy generated by moving cosmic string can be written as:

$$\frac{\delta T}{T} = 4\pi G\mu\beta \left(\pm 1 - \frac{\Phi_r}{\pi}\right) \frac{\cos\phi \sin\theta}{1 - \beta \sin\phi \sin\theta}, \quad (1)$$

where

$$\tan\phi_r = \frac{\sin\phi \sin\theta}{\cos\phi \sin\theta - R}.$$

Therefore, the anisotropy induced by a single cosmic string shows a sequence of zones of decreased and increased temperature in a pattern which, moving toward the string can be schematized as a cold spot right in front of the moving string followed by a step-like jump across the

string and then by a hot spot immediately after the string and, finally, another cold spot.

The amplitudes of these discontinuities and their extensions depend on the position of the string with respect to the observer, on the string velocity and its direction, and on the string linear density, but the structure is always the same. Simulations show that for cosmic strings with deficit angles from 1 to 2 arcsec (i.e. corresponding to the linear density associated to the GUT energy scale) the amplitude of the anisotropy varies between 15 and 30 μK. The upper bound on the string deficit angle is $D = 6''$, which corresponds to $\delta T/T \approx 100\mu$K and is determined by the fact that for such value the anisotropy induced by the cosmic string is comparable to that induced by adiabatic fluctuations and therefore it should have been detected directly. In our work we neglect the curvature which would be induced in the observed string by the time delay of the photons originating at different points of the string [14].

III. Simulations

In order to study the effects of a cosmic string on the CMB, we created a robust set of maps of sky simulating the signal and noise structure of WMAP data. The maps are divided into two datasets: "Only CMB" and "Only String". The "Only CMB" set is composed by 300 maps not containing the signature of the string. These maps were produced starting from a simulated power spectrum, generated by CMBEASY [7], a lighter and faster version of CMBFAST [8].

The "Only String" set is instead obtained by means of a C++ code specifically written to simulate the distribution of temperature fluctuations induced by a straight cosmic string. According to eq.(1), in order to define the string features, we used 3 parameters (Fig. 1):

1) the magnitude β of velocity (set c=1);
2) string direction κ, defined as the angle between the velocity vector and the line of sight;
3) distance ξ between the observer and the string, setting the distance observer-last scattering surface equal to 1.

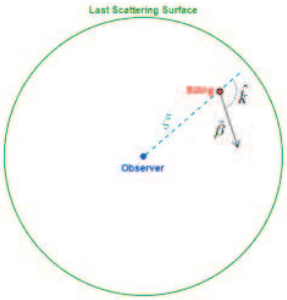
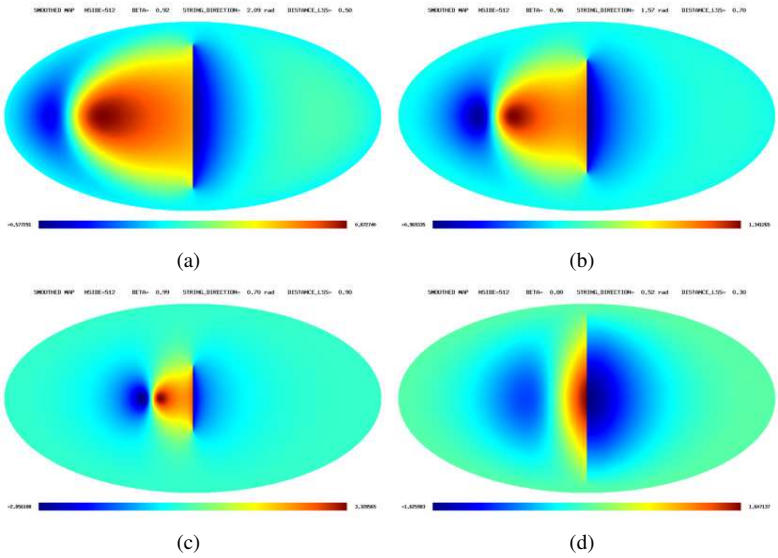


Fig. 1. String parameters used for the simulations.

The simulations were generated using the HEALPix package ([9]), in order to create the map and perform a multipole analysis. Due to its close relation with HEALPix, this code can be easily interfaced with CMBFAST and other tools for CMB analysis. It is worth to stress that this code is able to produce only straight strings and does not consider the effect of retardation of photons. In Fig. 2 the flowchart of our code is shown: the user sets string parameters and then the code generates the temperature map, in according to eq.(1). Using the HEALPix package, the map is stored in a file, a TGA image is created, and successively a median filter of previously defined radius (to simulate the finite resolution of the detector) is performed, then the smoothed map and its TGA image are stored in another file. The subsequent step is a multipole analysis to generate the files of a_{lm}s and C_ls.

TABLE I
Parameters used for string simulations

β	κ	ξ
0.0 ÷ 0.9	0 ÷ 180	0.0 ÷ 0.9
0.920		
0.940		
0.960		
0.980		
0.985		
0.990		
0.992		



(a) (b) (c) (d)

Fig. 3. Examples of simulations of straight cosmic string signature in the CMB for different values of parameters: a) $\beta = 0.92$, $\kappa = 120°$, $\xi = 0.50$; b) $\beta = 0.96$, $\kappa = 90°$, $\xi = 0.70$; c) $\beta = 0.99$, $\kappa = 40°$, $\xi = 0.9$; d) $\beta = 0.80$, $\kappa = 30°$, $\xi = 0.30$.

Fig. 2. Flow-chart of the string signature simulation procedure. Starting from the string parameters, a dT/T map is generated and the harmonical analysis is performed. For each simulation a map file, an image file, and a_{lm}s and C_ls are created

By varying the string parameters according to what is shown in Table I, we obtained different moving strings with their own signature of the CMB (Fig. 3). In order to reduce the computational time and to produce a large number of simulations, we were obliged to recur to 64 processors of the GRID infrastructure of PON S.Co.P.E.. In about 6 hours of CPU time 3040 simulations of maps were generated and 18240 files were saved on a storage element of the GRID for a total of about 100 GB of compressed data.

The two types of data ("Only CMB" and "Only String") were combined in "String + CMB" maps to produce realistic simulated images sampling the whole parameter space defined by S/N (fluctuations due to cosmic string/CMB) and by the string parameters. To do this, a separate routine of the code adds the pure string map to the pure CMB map to get a CMB map with superimposed string, using the following relation:

$$\delta T_{CMB+String}(i) = \delta T_{CMB}(i) + \alpha \delta T_{String}(i)$$

where i represents the pixel index and α is a multiplicative coefficient strictly related to the S/N ratio. Varying α, in fact, we obtain different

values of S/N ratio, in according to Table II.

TABLE II
TABLE OF CONVERSION α - S/N RATIO

α	**S/N Ratio**
1	0.002
5	0.01
10	0.02
15	0.03
20	0.04
30	0.06

In Fig. 4 an example of "String + CMB" map is shown, in which, in order to stress the effects of cosmic string, we used a high S/N ratio ($\alpha = 100$).

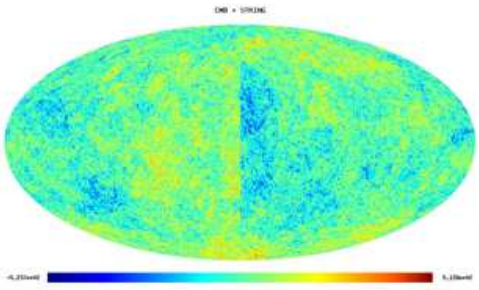


Fig. 4. Example of "String + CMB" map with $\alpha = 100$ (S/N ratio 0.2). CMB is been generated by CMBEASY. Parameters of string are: $\beta = 0.70$, $\kappa = 90°$, $\xi = 0.60$.

IV. THRESHOLD FILTER

As it was stressed in the introduction, our main goal is to see what are the upper limits to detect cosmic string signatures in CMB data. At present there is no evidence of their presence, and the data gathered by modern space missions (such as WMAP) show that, if the strings do exist, their contribution to the total anisotropy is very low. This implies that the S/N ratio between the string signal and the CMB anisotropy is very low. Therefore, in order to recognize the possible signal introduced by a cosmic string, we need to create a filter finely tuned to work with very low S/N ratios. The filter we propose in this paper is the first step to the realization of a more complete and efficient filter that could recognize string signatures regardless of the string shape and orientation in the sky. This filter consists of two steps:

1) the data are elaborated in order to reduce noise and amplify signal: this procedure is called "squeezing";
2) a trigger with defined threshold is applied to new data.

A. The "squeezing" procedure

The "squeezing" procedure allows to damp the noise due to the CMB anisotropy and to amplify the string signal, so that the step like hot-cold spot, which is characteristic of a string, becomes more easily detectable. This procedure makes a partition of the sphere (then of the map) in strips of $\Delta\theta = 10°$, centered on the equator. In each strip, for fixed longitude ϕ, the average $\delta\overline{T}$ of the anisotropy values over the latitude θ is calculated, so that the map is reduced to a monodimensional vector of 4096 pairs $(\phi, \delta\overline{T}(\phi))$. Finally all strips are summed together. In Fig. 5a and 5c squeezed "Only String" and "String + CMB" maps with $\alpha = 20$ respectively are shown. In both cases the step-like discontinuity appears clearly visible.

B. The trigger

A preliminary version of the trigger was implemented above all to evaluate the recovering capabilities of the squeezing procedure. For each squeezed map the mean value $\overline{x}$ and the standard deviation σ are evaluated. Then the code searches for a preset number of consecutive pixels that are above the threshold, which is defined as:

$$x_{thres} = \overline{x} + 2\sigma.$$

V. STATISTICAL ANALYSIS

In order to evaluate the efficiency of the "threshold filter", a statistical analysis was performed. In this analysis, we evaluated two characteristics of the filter: its capability to detect

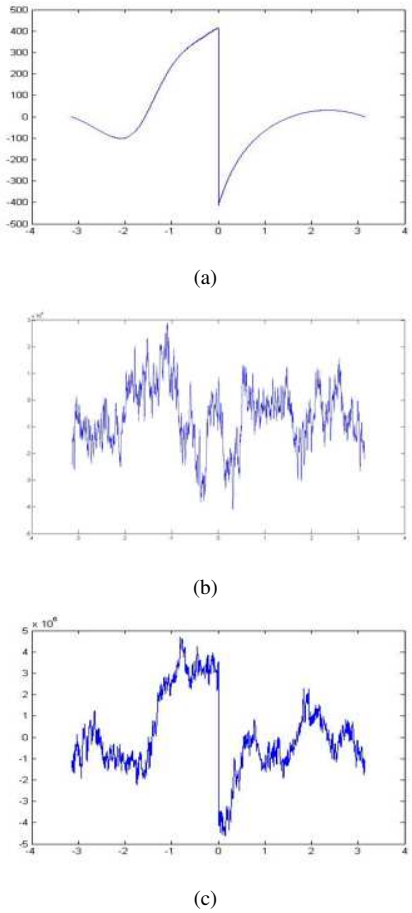


Fig. 5. a) Squeezed "Only String" map; b) squeezed "Only CMB" map; c) squeezed "String + CMB" map with $\alpha = 20$.

cosmic string signatures in the CMB anisotropy when it is present, and its cleanliness, that is the tendency of the filter to detect spurious events (i.e. over-threshold events due to CMB anisotropy and noise). To this purpose, we used two different datasets, created using the GRID infrastructure of PON S.Co.P.E.:

1) 300 "Only CMB" maps, simulating WMAP data (obtained using CMBEASY). These maps can be considered as noise for our purposes;
2) 1800 "String + CMB" maps, obtained combining the "Only CMB" maps with a template cosmic string (parameters $\beta = 0.70$, $\kappa = 90°$, $\xi = 0.60$) and varying the S/N ratio according to Table II.

"Only CMB" maps were used to evaluate the cleanliness of the filter. In fact, each triggered event of "Only CMB" maps represents a failure of the filter. Other important information about the "cleanliness" can be derived from the analysis of "String + CMB" maps. In fact, even if the primary result of "String + CMB" analysis is the capability to detect the cosmic string signatures, it is possible to estimate how many spurious events are encountered when there is also a string, well knowing that the event caused by the string must be unique (in analysing a map, the filter must detect only one event and all the other must be regarded as spurious). In order to evaluate the best filter configuration, the statistical analysis was performed also by varying, on both types of maps, the number of consecutive pixels required to be above the threshold and the basic parameter of the filter.

VI. Results and Conclusions

In Fig. 7 we present the results of statistical analysis on "Only CMB" data. In this case, the filter works well if no event is detected. By increasing the number of required consecutive pixels, the filter efficiency grows up, reaching a maximum (79%) for 50 consecutive pixels. In this case, however, the filter detects spurious events in 21% of cases.

TABLE III
Results of statistical analysis on "String + CMB" data for different S/N ratio

S/N	Best consec. pixels	Best Efficiency	Efficiency at 50 pixels
0.002	10	0.813	0.247
0.01	10 ÷ 33	1.000	0.810
0.02	10 ÷ 50	1.000	1.000
0.03	10 ÷ 50	1.000	1.000
0.04	10 ÷ 49	1.000	0.883
0.06	10 ÷ 23	1.000	0.009

In Fig. 6 the results of statistical analysis on the "String + CMB" data for different S/N ratio are shown and in Table III they are summarized (50 consecutive pixels above threshold, i.e. in

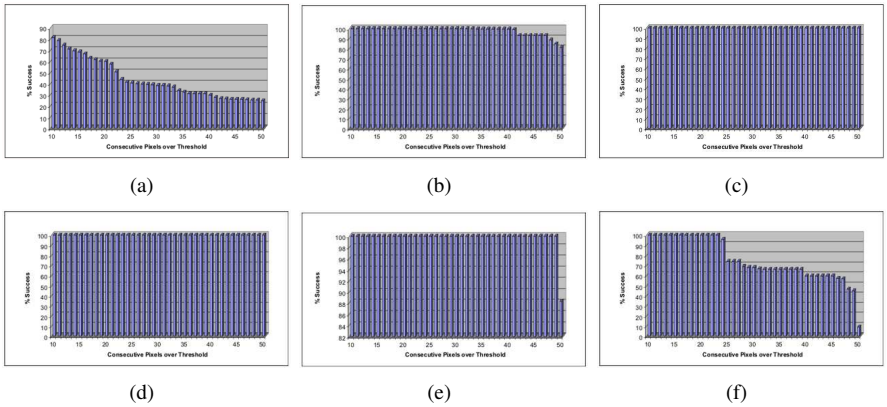


Fig. 6. Results of statistical analysis on "String + CMB" data for different values of S/N ratio: a) S/N=0.002; b) S/N=0.01; c) S/N=0.02; d) S/N=0.03; e) S/N=0.04; f) S/N=0.06

the best case for "Only CMB" data described above) are shown.

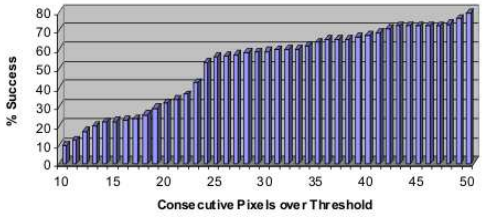


Fig. 7. Results of the statistical analysis performed on "Only CMB" maps as a function of the number of required consecutive pixels above threshold. The filter must be regarded as successful if no event is detected. The best filter efficiency of 0.79 (i.e. 0.21 of spurious events detected) is reached for 50 consecutive pixels.

It is apparent that the filter works well for S/N ratios between 0.01 and 0.04. At lower S/N it fails because the signal due to the string is too low, while at higher S/N, it fails since the threshold becomes too high. In addition, we have to consider that the capability of the filter of not recognizing spurious events in presence of a string is very low. In many cases, expecially when the number of required consecutive pixels is low, the incidence of spurious events is so high to render the filter practically useless.

In order to reduce the incidence of spurious events and simultaneously increase the efficiency in both cases quoted above, we are developing 2 new filters: 1) an upgrade of the filter just described; 2) an adaptive filter capable to recognize the scale lenght of the local signal and the shape of the string signature. Both filters need huge computational resources, so they will foresee an intensive use of GRID infrastructure.

ACKNOWLEDGMENT

This work was funded by the S.Co.P.E. project, and by the Italian Ministery of Foreign Affair (MAE) through a bilateral (Italy-USA) agreement.

REFERENCES

[1] Kibble T.W.B., J.Phys. A: Math. Gen. V.9, 1387, 1976
[2] Zeldovich Ya.B., 1980, MNRAS, 192, 663
[3] Vilenkin A., Phys. Rev.,D, 23, 852, 1981
[4] Davis A.C. & Kibble T.W.B., Contemporary Physics, Vol. 46, 313, 2005
[5] Polchinski J. & Rocha J.V., Phys. Rev. D 75, 1235, 2007
[6] Vilenkin A. & Shellard E.P., Cosmic strings and other topological defects, Cambridge Univ. Press., 1994
[7] http://www.thphys.uni-heidelberg.de/∼robbers/cmbeasy/
[8] http://www.cfa.harvard.edu/∼mzaldarr/ CMB-FAST/cmbfast.html
[9] http://healpix.jpl.nasa.gov/
[10] Kaiser N. & Stebbins A., Nature, v.310, p.391, 1984
[11] Gott J. R., ApJ, 288, 422, 1985
[12] Jeong E. & Smoot G.F., ApJL, 661, L1, 2007
[13] O.S. Sazhina, M.V. Sazhin & V.N. Sementsov, JETP, v. 133, I. 5., 1005, 2008
[14] A. Vilenkin, Nature, 322, 14, 1986.

Riccio Giuseppe (Napoli, Italy, June 16, 1978) studied Physics at University of Naples ”Federico II” and graduated in Astrophysics within the framework of Nemo Project with the thesis: ”Studio di fattibilità di un algoritmo di trigger di primo livello per l’astronomia di neutrini cosmici da sorgenti astrofisiche con l’esperimento NEMO” in 2006. At present he is a Ph.D. student in Fundamental Physics and Applications of University of Naples ”Federico II” working on research for cosmic strings and neural networks for photometric redshifts evaluation.

D’Angelo Giovanni (S.Maria Capua Vetere (CE), Italy, April 2, 1974) studied Physics at University of Naples ”Federico II” and graduated in Astrophysics within the framework of Nemo Project with the thesis: ”Studio di un algoritmo di ricostruzione di eventi muonici da neutrini cosmici per il telescopio sottomarino NEMO” in 2006. At present he is a fellow in the S.Co.P.E. Project of University of Naples ”Federico II” working on Data Mining on Massive Data Sets, applications porting on GRID infrastructure and research for cosmic strings.

Overview of the Contributions of the LHC Experiments in INFN GRID for Bringing the Grid to Production Quality

Attilio Andreazza[18], Giuseppe Bagliesi[9], Stefano Bagnasco[1], Roberto Barbera.[7], Luciano Barone[10], Stefano Belforte[11], Michela Biglietti[20], Massimo Biasotto[8], Tommaso Boccali[9], Daniele Bonacorsi[12], D. Bortolotti[12], Simone Campana[16], A. Carbone[12,20], Gianpaolo Carlino[20], Piergiorgio Cerello[1], Claudia Ciocca[4], Mattia Cinquilli[13], Giuseppe Codiposti[12],Andrea Dainese[8], Barbara Dalena[3], Alessandro De Salvo[10], Domenico Di Bari[3], Alessandro Di Girolamo[16], Giacinto Donvito[3], Alessandra Doria[20], Alessandra Fanfani[12], Federica Fanzago[14], Fabio Farina[15], Maria Lorenza Ferrer[17], Domenico Galli[12,20], Pietro Govoni[15], Claudio Grandi[12], Stefano Lacaprara[14], Lamberto Luminari[10], Stefano Lusso[1], Giorgio Maggi[3], Nicolo' Magini[16], U. Marconi[12], Agnese Martini[17] , Massimo Masera[2], Alberto Masoni[5], Leonardo Merola[20], Vincenzo Miccio[4], Elisabetta Molinari[18], Daniele Mura[5], Guido Negri[16], Francesco Noferini[4], Marco Paganoni[15], S. Perazzini[12,20] , Laura Perini[18,19], Giuseppe Platania[6], Alberto Pulvirenti[7], David Rebatto[18], Silvia Resconi[18], Lorenzo Rinaldi[4], Leonardo Sala[15], R. Santinelli[16], Subir Sarkar[9], Andrea Sartirana[4], Andrea Sciaba'[16], Daniele Spiga[16], Vincenzo Spinoso[3], Luca Vaccarossa[18], V. Vagnoni[12], Elisabetta Vilucchi[17]

[1]INFN, Sezione di Torino, [2]Dipartimento di Fisica Sperimentale dell'Universita' di Torino,
[3]Dipartimento di Fisica dell'Università e INFN, Bari, [4]INFN-CNAF, Bologna,
[5]INFN, Sezione di Cagliari, [6]INFN, Sezione di Catania,
[7]Dipartimento di Fisica dell'Università e INFN, Catania,
[8]INFN, Laboratori Nazionali di Legnaro, [9]INFN, Sezione di Pisa,
[10]INFN, Sezione di Roma1, [11]INF N, Sezione di Trieste, [12]INFN, Sezione di Bologna,
[13]INFN, Sezione di Perugia, [14]INFN, Sezione di Padova,
[15]INFN, Sezione di Milano Bicocca, [16]CERN, Organizzazione Europea per la Ricerca Nucleare,
[17]INFN, Laboratori Nazionali di Frascati [20] INFN, Sezione di Napoli,
[18]INFN, Sezione di Milano, [19]Dipartimento di Fisica dell'Università' di Milano,
[20] Dipartimento di Fisica, Università di Bologna

Abstract — **The Italian groups participating to the LHC experiments with INFN funding, have given very substantial contributions in bringing the European Grid Infrastructure to the high level of reliability and efficiency that has finally been reached. This research and development work was performed by the members of the computing groups of the experiments, that coordinated themselves within the INFN Grid project. In this paper we present an overview of the Data Challenges the experiments have performed on the Grid Infrastructure with increasing complexity and involvement of sites and resources and highlight the results achieved in steering the grid middleware and operations to real maturity for use in production. The crucial role played by INFN Grid members in the experiments is also illustrated together with the working of the organisms that have allowed feedback from the computing activity of the LHC experiments to be efficiently incorporated in the grid infrastructure and the main steps and results thus achieved.**

***Index Terms* — LHC, Nuclear Physics, Grid.**

I.INTRODUCTION

During the past few years the Italian LHC experiments computing teams, organized in the INFN Grid project, have given very substantial contributions for bringing the European Grid infrastructure to the high level of reliability and efficiency that is needed for sustaining the production of the huge amount of data these experiments deal with, for simulation, reconstruction and analysis. A summary of the evolution of the usage of the Grid by the

experiments and of the role played by INFN actors in this evolution is given separately for each of the four experiments, and general conclusions are presented at the end

A.The ATLAS Experiment and the Grid

1)The initial phase

The first large scale world-wide exercise of ATLAS on the Grid was Data Challenge 2, performed between May and December 2004: in DC2 about 10 M events were simulated using a production system capable of submitting jobs to the 3 different Grids present in ATLAS: EGEE-LCG, Nordugrid, OSG (via 3 different executors, developed in ATLAS). The number of long jobs (typically 20 hour per job) running per day reached about 2500-3000, of which more than half on EGEE-LCG, in total 235000 jobs were processed, and 6 TB of data were transferred.

The Data Management system was at the very beginning: the ATLAS layer Don Quijote was at its first real test, the Globus Replica Location Service (RLS) performed not well enough, the Information System was unstable, and the Resource Broker was unable to sustain the job load for long enough [1].

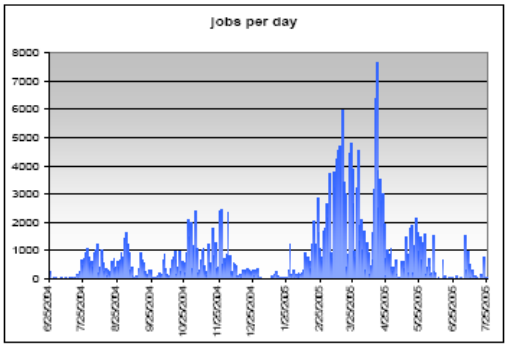


Figure 1. Number of jobs run every day on the native LCG-2 infrastructure during DC2 (July04 to January05) and Rome Production (January05 to June05)

ATLAS then invested much effort in ameliorating the Grid middleware, via testing and interaction with the developers, as well as the ATLAS layer. The improvements were very relevant, as were evident in the so called "Rome Production" run.

This production was performed for providing the data for the Physics Workshop in Rome: 380 k Jobs were submitted on LCG native site, using the WMS (evolution of RB); 1.4 M files were stored for a total 45 TB of data. INFN site had contributed for 25% to the production and the people manning it were INFN for > 50%. As shown in the Table, the problems of Information System were largely solved (with the introduction of BDII), together with the job brokering ones. The major sources of troubles were now the Data Management and the sites.

System	Causes	Rate
WMS	Total	1.6%
DM	Download input files	26.4%
DM	Upload output files	0.8 %
ATLAS/LCG	Athena-crash	9.1%
ATLAS	Proxy Expired	0.3%
LCG	Site misconfig.	0.9%
Unclassified		9%

Table 1. Causes of job failures during Rome Production.

2)The efforts for improvement and the recent production results

The efforts of ATLAS and LCG-EGEE teams have continued and the collaboration between the experiments and these teams have been strengthened. The creation since the end of 2005 of the Technical Coordination Group, in which the leaders of the LCG-EGEE team sit together with the leaders of the Task Forces the experiments had created for bringing the LCG-EGEE to full production level. In ATLAS the TCG representative was from the beginning an INFN physicist (Laura Perini), and the role was then taken up by a CERN one that had been associated with INFN (Simone Campana). The results achieved are documented in the papers presented at conferences in 2006 and 2007 [3], [4].

The list of the signatures of the four paper referenced shows clearly how central the INFN role has been in this effort.
Figure 2 shows the level reached by the ATLAS productions in 2008; the mean level of 12k jobs is maintained for long time, with peaks of >25 k.

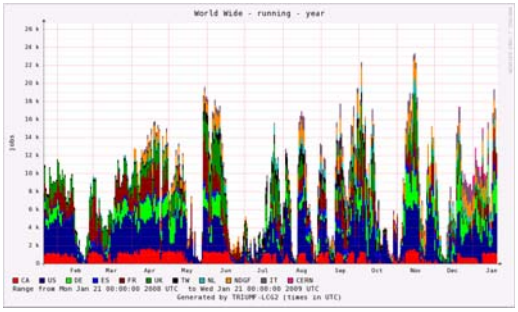


Figure 2. ATLAS Production jobs in 2008.

About 2/3 of the work is done in LCG-EGEE sites. The main structures are due to release schedule, validation latency, modest requests for simulation (also connect to disk shortage). The wall clock time efficiency is > 80%, but the pilot based production system, introduced > 1 year ago in ATLAS, as well as the site monitoring and control, has still room for improvements.

3)The status of Data Management

The Data management has now basically reached the level needed for handling the expected load of real data.
In January 2009 the 10M file test was performed, in order to verify such readiness, in which each Tier0,1 sites had to import 1M files from the other centers disk storage in 10 days. The results in number of files for site per day are shown in Figure 3.

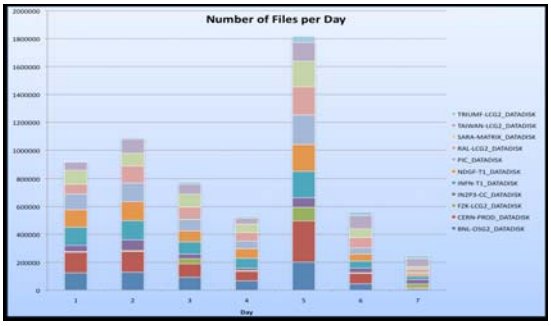


Figure 3. Number of files per day and site in the 10M files test.

Figure 4 shows the data completion rate after 10 days. The global ATLAS DDM system works well: some localized problems are visible in specific sites; for most of them the solutions have already been implemented and a new round of tests in April will verify them.

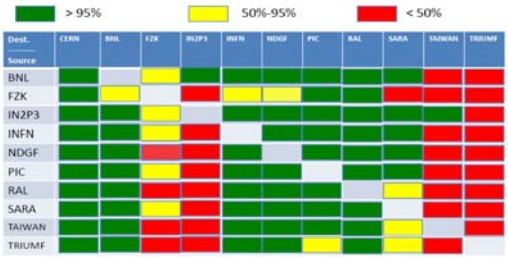


Figure 4 Data completion rate in 10M file test

B.The CMS Experiment and the Grid

1)The site readiness

CMS developed its own data placement agent, PhEDEx [5], which allows to generate data traffic transfer among sites for debugging and commissioning purpose.
A detailed procedure was defined in order to extensively test all relevant aspects of a Grid site in a continuous mode and integrated it into the daily operations. This procedure is needed to guarantee that data processing can be performed in an efficient and reliable way. The Debugging Data Transfer (DDT) task force uses the PhEDEx LoadTest tool to commission links between all CMS Tier centers, including

crosslinks between Tier-1 sites and upload/download links between Tier-1 and Tier-2 sites; automatic functionality and reliability tests (SAM test) are used in order to constantly monitor the quality of the TierN centers.

2) The use of the INFN Grid infrastructure

As can be seen from Fig. 5 and 6, the INFN Grid infrastructure is used continuously with good performance by final users. In particular in the Tier-2 sites both production of simulated events and end-user's analysis are taking place. It is now standard to see more than 100k concurrent CMS jobs running worldwide.

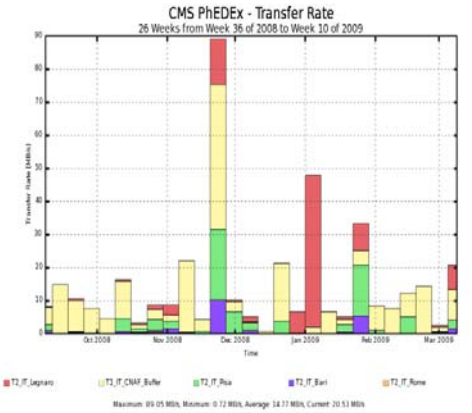


Figure 5: Data transfer to the INFN Tier-1 and Tier-2 centres.

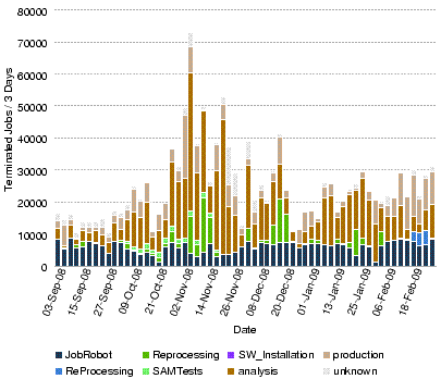


Figure 6: Terminated jobs on the INFN Grid infrastructure

This proves the stability and the maturity of the INFN-GRID infrastructure.

More than 10k jobs per day using only the Italian resources can be achieved.

3) CCR08

The Common Computing Readiness Challenge (CCRC'08) was carried out in order to test the readiness of the Worldwide LHC Computing Grid when all of 4 LHC experiments use their resources concurrently. During its second phase, in May 2008, the experiment workflows were exercised at the full scale foreseen for the startup of the data taking using a nearly final version of the deployed middleware. All the goals were reached successfully. An example is shown in Fig. 7, for the aggregated throughput between CNAF and the other regions

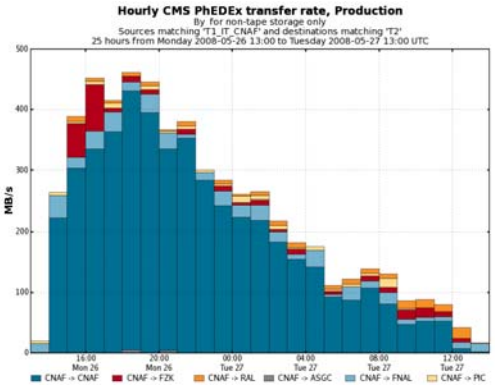


Figure 7: Data transfer from CNAF to the other regions during CCRC08

4)The data analysis at the Tier-2 sites

CRAB (CMS Remote Analysis Builder) [6] is a specific tool, designed and developed by INFN, that allows a transparent access to distributed data to end-user physicist. The tool adopted a client-server architecture in order to automate the analysis workflow, leaving to the user just minor actions. In this way CRAB fully integrates into WLCG Grid infrastructure and into the CMS data and workflow management system. Peaks of daily submissions equal to 70000 jobs/day, for a

total of 10 million jobs in 2008, on simulated data and real cosmic data, have been reached. The number of distinct CRAB users is about 1000, which is approaching the 50% of the whole CMS community and more than 100 users submit jobs each day.

C.The ALICE Experiment and the Grid

The ALICE experiment at CERN is a general-purpose heavy-ion experiment designed to study the physics of strongly interacting matter and the quark-gluon plasma in nucleus-nucleus collisions at the Large Hadron Collider. ALICE currently uses Grid Computing for all of its needs of data reconstruction, batch analysis and Montecarlo simulation productions, and plans to extend this also for distributed interactive analysis [7].

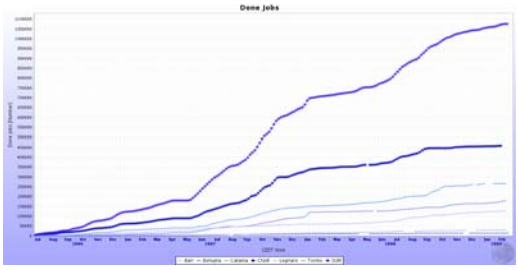


Figure 8. Number of completed ALICE jobs on the INFNGRID infrastructure.

ALICE has been pursuing a fully Grid-based computing model since the very beginning by developing a proprietary full-fledged grid middleware [8] that later evolved into a quite complex system leveraging on most of the available Grid services, complementing them with some extra functionalities. The first solution was a simple "pull-model" submission system, in which remote Computing Elements requested jobs from a central Task Queue, while the current schema involves Job Agents running on Worker Nodes that continuously interact with services, building a "virtual private Grid" on top of the existing infrastructure.

The system is interfaced with site Grid Services by means of a dedicated server, named VO-Box, that runs thin interfaces to local CE and SE services as well as specific experiment services like Software Package Management. While in the first implementations these interfaces were geographical in scope, it was found that a much better integration, particularly for monitoring tasks, was achieved by having some services running locally to each site.

Using this approach, it was possible to move from a system useful only for Montecarlo productions to a more complex system (the version used for PDC04 in 2004) that allowed a two-phased production in which background events generated in Phase I were repeatedly merged with different signal events and reprocessed [9]. This strategy is peculiar to heavy ion experiments, in which physical interesting processes are superimposed to a background comprising several thousands of tracks. Such background events are computationally expensive to simulate and are re-used about ten times with different signal events.

Subsequently the adoption of xrootd [10] as access protocol, and its integration with standard Grid

Figure 9. Completed ALICE jobs from single users (unscheduled activity). Different colors refer to different individual users accessing Grid services.

Storage Element architectures like CASTOR2, dCache or DPM, allowed simple and uniform access to data locally and remotely both for Grid jobs and interactive processing.

Following this development, single users started routinely using the Grid for Montecarlo productions and batch analysis.
The ALICE computing model envisages also interactive unscheduled ("chaotic") access to data for the last stages of analysis. In order to provide the experiment community with parallel resources for such activities ALICE adopted PROOF (Parallel ROOT Facility) an extension of the ROOT system and deployed an Analysis Facility at CERN [11].
An INFN lead project is currently developing a tool to deploy such parallel interactive facilities on existing Worker Nodes in Tier-2 centers, by means of advanced virtualization techniques [12].
ALICE has been exercising its Grid infrastructure with periodical Physics Data Challenges (PDC) since 2002. After PDC06, ALICE has been running in nearly continuous mode, completing almost 1.1 million jobs on the INFNGRID infrastructure alone.

D.The LHCb Experiment and the Grid

LHCb uses the SAM framework for extensively test all the relevant aspects of the WLCG sites:

- availability and health status of the Computing Elements;
- type of the Operating System and architecture; availability of all the SRM instances;
- availability of the data links (it mainly concerns testing interconnections among CERN and Tier-1s);
- to install appropriate versions of the software in the shared areas.

In order to submit jobs (in pull mode) to sites of the WLCG, LHCb relies on the DIRAC framework (Distributed Infrastructure with Remote Agent Control) [14]: it is the experiment gateway to the Grid and its key-words are traditionally: resilience, reliability and redundancy. DIRAC provides Data Management Tools which allow managing data transfers among the primary Grid source of storage of CERN and of Tier-1 centres (Tier-2 are also foreseen in LHCb but only to produce simulated events, being equipped then with minor storage capabilities).
Over the past few years, LHCb has always been using WLCG resources, gathering considerable experience in distributed computing at a large scale. CCRC'08 represented a crucial benchmark being the last chance of testing services and resources prior real data come. Its main goals were both to consolidate the LHCb Computing Model, and testing its newly commissioned computing framework (DIRAC3); as well as exercising and testing the readiness of all resources and services involved in the Grid, by having all the four LHC experiments running concurrently.
Data Management System and Workload System on WLCG have been proved to be stable enough for sustaining the LHCb activities. A more general feeling that the underlying infrastructure had scaled with the needs of the four HEP VOs running their activities at full nominal rate, is also perceived. The reliability of all these services has improved a lot since the past.
This has been certainly possible by instituting adequate operational infrastructures closely watching and chasing problems and also by means of progressed communication channels between VOs and resources and service providers. A significantly advanced monitoring infrastructure as well as improved procedures could also be assessed as big contribution to this success. CCRC'08 also highlighted some still pending issues from the past. There is room for improvement but a decisive progress can be tracked down.

- during the CCRC'08 LHCb performed an exhaustive test of his computing model [13]: transmission of the RAW data from the pit to the Tier-0 (at the expected flux of the order of 60 MB/s);
- RAW data distribution from the Tier-0 to the Tier-1s (about 10 MB/s each);
- fast reconstruction and pre-selection (stripping) of the RAW data at CERN and Tier1s, to build up the reduced data samples in the rDST format;

- production of data in the DST analysis format (starting from rDST and RAW), for distributed data analysis at CERN and Tier1s;
- distribution of the DST copies to all the Tier-1 centres (DST are made available at each Tier-1 centre for data analysis).
- CNAF, as planned, contributed to CCRC'08, with high reliability to about 10% of the overall processing activity.

Fig.10 shows the breakdown of the data transfer from the Tier-0 to the LHCb Tier-1s during the CCRC'08. Fig.11 shows the amount of jobs processed by means of the Italian Grid during the last year: something like 3.7×10^5 jobs processed in total: mainly at the LHCb CNAF Tier-1 but also accessing external resources. Fig.12 shows the progress made in the last year improving LHCb exploitation of the CPU resources: the plot shows the improvements achieved due to the performance of the file system used to effectively export to the worker node the software area.

The LHCb distributed analysis model foresees the usage of the Tier-1 centres only where data in DST format are available. GANGA [15] is used as the user interface to build up and to submit analysis jobs to the DIRAC backend. DIRAC in turn manages to effectively submit the job to the Grid acting as a centralized accounting and job prioritization system.

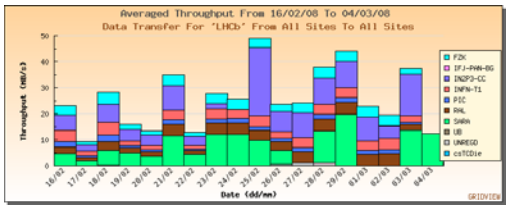


Figure 10: LHCb data transfer from the Tier-0 to theTier-1s during CCRC'08

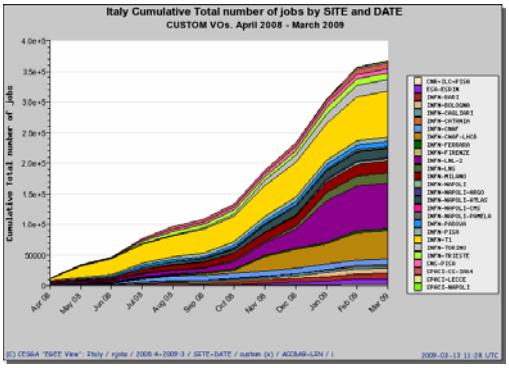


Figure 11: LHCb data transfer from the Tier-0 to theTier-1s during CCRC'08

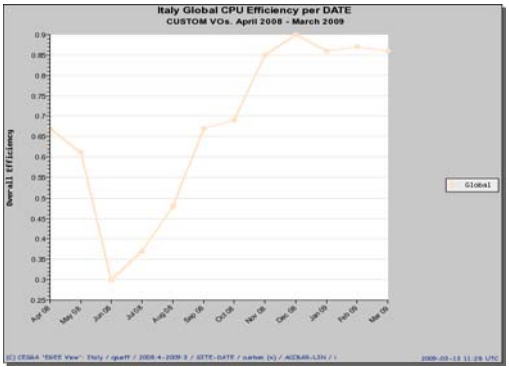


Figure 12:CPU efficiency for job submission in the Italian Grid by LHCb

CONCLUSIONS

During the past five years the reliability of the INFN Grid has steadily improved, thanks to an improved middleware and a better knowledge by the administators of Tier-1 and Tier-2 centres. At the same time the available resources and their use from the LHC experiments have significantly increased, as clearly visible in Fig. 13.

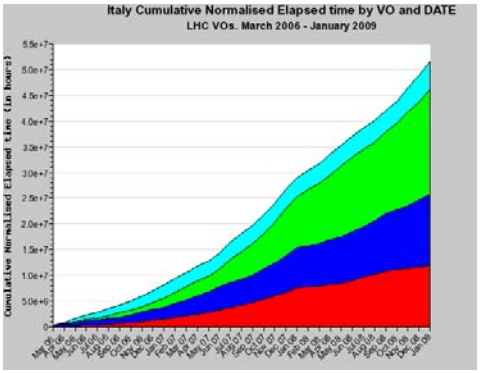


Figure 13: Cumulative cpu time used on INFN-Grid by the four LHC experiments (ALICE in red, ATLAS in blue, CMS in green and LHCb in azure)

REFERENCES

[1] González de la Hoz, S et al. "ATLAS Data Challenge 2 : A massive Monte Carlo production on the GRID" ATL-SOFT-PUB-2005-001; ATL-COM-SOFT-2005-003.- Geneva : CERN, 2005 . - 10 p.

[2] S. Campana et al. "Analysis of the ATLAS Rome production experience on the LHC computing grid" In: Proceedings of the first international conference on e-Science and Grid Computing : Melbourne, Australia, December 5-8, 2005: IEEE Computer Society, 2005. - ISBN 0-7695-2448-6. - p. 82-89.

[3] C. Aiftimiei et al. "Prototyping production and analysis frameworks for LHC experiments based on LCG/EGEE/INFN-Grid middleware" In: Computing in high energy and nuclear physics : CHEP-2006 : vol. 2 ; Sunanda Banerjee. - New Delhi : MacMillan, 2007. - ISBN 0230-63017-0. - p. 998-1001. Computing in high energy and nuclear physics : CHEP, Mumbai, 2006.

[4] X. Espinal et al. "Large-scale ATLAS simulated production on EGEE" In: Proceedings of the third IEEE international conference on e-science and grid computing ; Geoffrey Fox, Kenneth Chiu, Rajkumar Buyya. - Los Alamitos, CA : IEEE computer society, 2007. - ISBN 0-7695-3064-8. - p. 3-10. (Intervento presentato al 3. IEEE International conference on e-science and grid computing, Bangalore, 2007).

[5] http://cmsweb.cern.ch/phedex/documents.html

[6] D. Spiga et al, The CMS Remote Analysis Builder (CRAB), Lect.Notes Comput.Sci., 4873:580-586,2007.

[7] The ALICE Collaboration, ALICE Technical Design Report of the Computing, CERN-LHCC-2005-018 (2005)

[8] S. Bagnasco et al. "AliEn2: ALICE Environment on the Grid" J. Phys: Conf Ser **119**:062012 (2008)

[9] L. Betey et al, "The ALICE Physics Data Challenge 2004 and the ALICE Distributed Analysis Prototype" in proceedings of 2004 Conference for Computing in High-Energy and Nuclear Physics, Interlaken, Switzerland (2004)

[10] A. Dorigo, P. Elmer, F. Furano and A. Hanushevsky "XROOTD/TXNetFile: a highly scalable architecture or data access in the ROOT environment" in proceedings of the 4th WSEAS international Conference on Telecommunications and informatics, Prague, Czech Republic (2005)

[11] J.F. Grosse-Oetringhaus, "The CERN Analysis Facility – A PROOF Cluster for Day-one Physics Analysis" J. Phys.: Conf. Ser. 119:072017 (2008)

[12] D. Berzano, S. Bagnasco, M. Masera, S. Lusso "A prototype of a dynamically expandable Virtual Analysis Facility" in Proceedings of ACAT08, Erice, Italy (2008)

[13] LHCb Computing Model, CERN/LHCb-2004-119

[14] The LHCb Data Production and Distributed Analysis System, A.Tsaregorotsev et al., Proc. 2006 Conference for Computing in High Energy and Nuclear Physiscs, Mumbai India 2006

[15] http://ganga.web.cern.ch/ganga/

Distributed Analysis for the ATLAS Experiment in the S.Co.P.E Project

G. Carlino[1], A. Aloisio[1,2], M. Alviggi[1,2], M. Biglietti[1,2],
V. Canale[1,2], F. Cevenini[1,2], G. Chiefari[1,2], F. Conventi[1,2,3],
D. Della Volpe[1,2], A. Doria[1], L. Merola[1,2], E. Musto[1,2], S. Patricelli[1,2]
[1] Istituto Nazionale di Fisica Nucleare - Sezione di Napoli, Italy
[2]Università degli Studi Federico II - Dipartimento di Scienze Fisiche, Napoli, Italy
[3]now Università Parthenope, Napoli, Italy

***Abstract*—ATLAS is a particle physics experiment that will operate, starting in spring 2009, at LHC (Large Hadron Collider) at CERN. LHC will achieve a luminosity of 10^{34}cm^{-2}s^{-1} and will be able to explore a large amount of particle physics fields. Its main goal is to detect Higgs boson, the only Standard Model particle yet to be observed. ATLAS is a large international collaboration, made up of more than 2000 physicists from all over the world. This complexity requires a very cooperative distributed computing and data analysis model, so the emergent GRID Computing technology has been adopted. ATLAS computing model has been designed in order to develop and keep operative a system capable to manage some PetaBytes of data per year, so the ATLAS experiment, in collaboration with two European projects, LCG (LHC Computing Grid) and EGEE (Enabling Grid for E-SciencE), is setting-up a complex set of tools and services for automated data distribution and processing. ATLAS software has been installed on the S.Co.P.E prototype and different integration tests have been performed to evaluate at best the physics analysis workability degree. On the bases of these results, the job submission on the prototype by using the ATLAS Virtual Organization, the data transfer on the Storage Element and the configuration of the ATLAS specific tools have been performed in S.Co.P.E environment; in particular, the distributed analysis tool GANGA has been set up and used.**

***Index Terms*—LHC, ATLAS, LCG, S.Co.P.E.**

I. Introduction

THE aim of this paper is to give an overview of ATLAS Distributed Computing Model and to explain how this model has been integrated in S.Co.P.E. Project.

II. LHC overview

The CERN Large Hadron Collider (LHC) [1] is a high energy proton-proton collider installed in the existing 26.7 km tunnel about 100 m underground near Geneva constructed between 1984 and 1989 for the CERN Large Electron Proton (LEP) machine. A magnetic field of 8.5 Tesla enables the two proton beams to travel, in opposite direction, through two superconducting rings, each of 4.2 km radius. Each beam, composed of 2835 bunches containing 10^{11} protons, will achieve a energy of 7 TeV, in order to reach, in the collision, a center of mass energy of 14 TeV. The crossing rate of 40 MHz will produce a collision rate up to 10^9 Hz. Starting with a luminosity of 10^{33}cm^{-2}s^{-1}, LHC will reach in three years a luminosity of 10^{34}cm^{-2}s^{-1}.

In the four proton-proton colliding points (see fig.1) are placed four experiments: two "*omni purpose*" experiments, ATLAS [2] and CMS [3], and two dedicated experiments: LHCb [4] for B-physics studies, and one heavy ion experiment, ALICE [5].

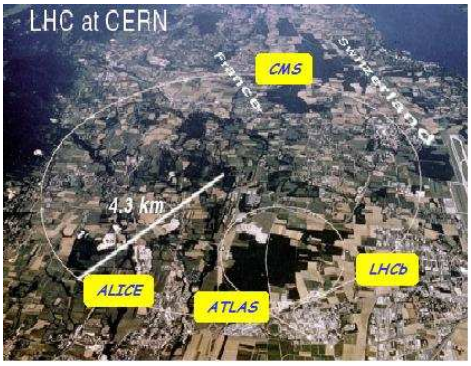


Fig. 1: LHC colliding points scheme.

A. LHC Physics Motivation

The unprecedent characteristics described above make LHC the largest high-energy physics project all over the world enabling the study in deep details of a large amount of particle physics items.

The model describing accurately elementary particles phenomenology is the Standard Model, formulated in 1967 by Glashow [7], Salam [8] and Weinberg [9]. In this model, describing for the first time both electromagnetic and weak interactions together, elementary particles are classified on the basis of the statistical laws they obey, distinguishable by their "intrinsic" spin angular momentum. Spin can be an integer number or a semi-integer number: in the first case particles are called "bosons", according to Bose-Einstein statistics, while in the second case particles are called "fermions", according to Fermi-Dirac statistics. Leptons and quarks are fermion particles, classified in three generations:

$$\text{Leptons:} \quad \begin{pmatrix} \nu_e \\ e \end{pmatrix} \begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix} \begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}$$

$$\text{Quarks:} \quad \begin{pmatrix} u \\ d \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} \begin{pmatrix} t \\ b \end{pmatrix}$$

These particles interact via fundamental forces, mediated by bosons particles; in particular the eight strong force mediators are called gluons, while electro-weak unified force is mediated by a massless particle, the photon, and by three massive bosons: Z^0, W^+ e W^-. Standard Model had a lot of experimental confirmations, but there are still some open questions. First of all, the determination of the origin of the electro-weak mass scale, operated by the Higgs boson [10], the only Standard Model particle yet to be observed. LEP experiments put a 95% C.L. lower limit on the Higgs mass at 114.4 GeV [11], and, in order to explore the whole mass range up to 1 TeV, a powerful machine such as LHC was needed. So the Higgs detection is the main goal of LHC research.

Moreover, the high value of the center of mass energy will allow LHC to search for new physics beyond the Standard Model, the more accredited model being Supersymmetry [12]. This theory [13] predicts the existence of new particles, "partners" of Standard Model particles, with equal properties but a spin differing by 1/2; this important characteristic establishes a new symmetry between fermions and bosons, i.e. between matter and force, and, if verified, will open new revolutionary fields in physics. In fact, Supersymmetry may solve some deep problems of the Standard Model and give an important answer to the Cosmology problem of the identification of the nature of Dark Matter.

It is remarkable to note that for a center of mass energy of 14 TeV, the total inelastic cross section is about 80 mb (see fig. 2), while for interesting processes such as Higgs or Supersymmetry particle productions the cross section is of the order of 1 pb. In terms of rates, the total inelastic rate is about 10^9 Hz, while Higgs production has a rate of the order of 1 Hz, which becomes about 10^{-2} Hz including the branching-ratio; this means that the selection power needed is of the order of 10^{11}. This is the real challenge of the LHC: to be able to identify and select such rare events in a background dominated by uninteresting events. For this purpose, very

large and complex experiments have been designed.

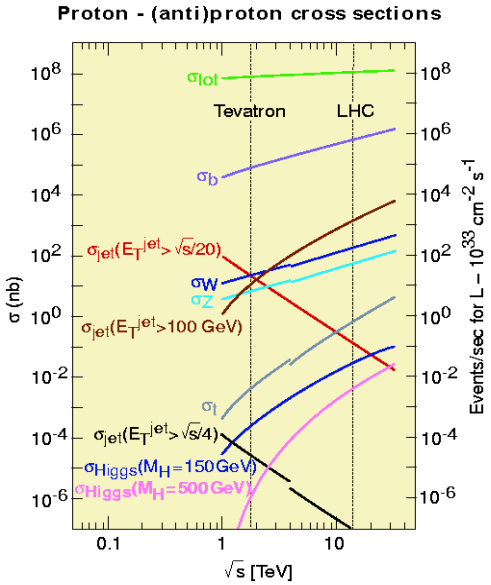


Fig. 2: LHC Cross Section.

III. ATLAS EXPERIMENT

ATLAS (A *Toroidal LHC ApparatuS*) is one of the four particle physics experiment at LHC, it is composed of more than 2000 physicists from hundreds of Research Institutes and Universities all over the world, and its layout has been set-up to exploit at best the LHC capabilities. Its technical characteristics will allow efficient reconstruction of charged particle tracks, photons and electrons identification and high precision measurements of muon momentum by using just the external muon spectrometer.

A. ATLAS layout

ATLAS experimental apparatus has a total volume of approximately 16000 m^3, a length of 46 m, a radius of 24 m, a weight of 7000 Tons and about 10^8 electronic channels. The layout [2] is shown in figure 3, where different regions can be singled out:

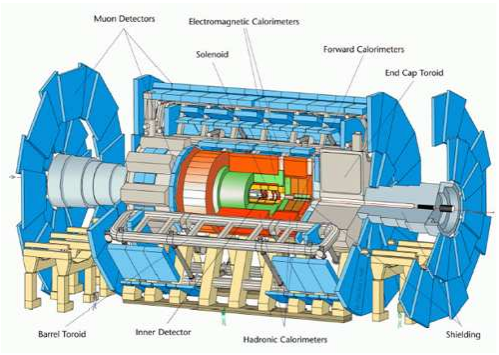


Fig. 3: ATLAS detector layout.

- the central region, called *barrel*, holds the sub-detectors, concentrically arranged compared to beams axis;
- two *end-cap* regions arranged at the ends of the *barrel*.

Proceeding form the inner to the outer, the *barrel* is composed by the following subdetectors:

1) the vertex tracker, surrounding the beam collision point;
2) the inner tracker, placed inside a solenoidal magnetic field;
3) the electromagnetic and hadronic calorimeters;
4) the muon spectrometer, placed such that particles which originate at the interaction point traverse three chamber stations inside a toroidal magnetic field.

In the *end-caps* are arranged, respectively:

1) the electromagnetic and hadronic calorimeters;
2) the forward calorimeters;
3) the muon spectrometer.

A schematic view of how particles are detected in ATLAS apparatus is represented in fig 4.

B. ATLAS Computing Framework

ATLAS software has been developed by using an *objected-oriented* model, based essentially on *C++* programming language, integrated with FORTRAN and Java components. A common *framework*, ATHENA [6], based on GAUDI [14], has been developed with an high

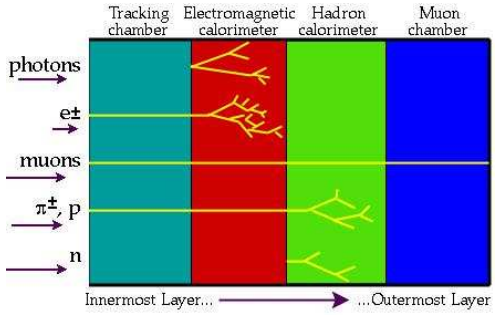


Fig. 4: Schematic view of particles detection in ATLAS

level of abstraction; it provides all infrastructures needed to process data from the detector read-out to data simulation, reconstruction, event display and analysis. ATHENA is based on a set of software interactive packages called components which operate different algorithms. Components are coordinated and supervised by an *Application Manager*.
The main feature of such framework is that user can define components to be used and how they must interact on the base of his needs by using Python scripts called *Job-Options*.
The different software versions are called *ATLAS releases* [24], physically organized using the Concurrent Versions System (CVS). Once a software release suitable for production has been built and validated, a distribution kit is used to install the ATLAS offline software on sites worldwide. The Package Manager used to fetch and install the ATLAS software is Pacman [15].

C. ATLAS Distributed Computing Model

High Energy Physics experiments experienced, during the years, an exponential growth from both hardware and software point of view. The large dimension of the ATLAS experiment and the complexity of the events requiring a huge granularity of the detectors in order to identify "good events" in a large background environment, determine a very large size of the events: 1.6 MB for Raw data.

Moreover, LHC beam collisions will take place with a temporal interval of 25 ns, and every bunch-crossing will produce, at high luminosity, about 23 interactions. Therefore the foreseen input event rate in the ATLAS detector will be of the order of 1 GHz. The ATLAS trigger, arranged in three levels, reduces such a rate to 200 Hz; this means that in one year the amount of data will be approximately of the order of 2 PB. Same amount of derived data will be determined by the large-scale processing, reprocessing and analysis activities. The computing power required to process this huge amount of data is of the order of 10^5 CPUs full time.

The impossibility of concentrating the needed computing power and storage capacity in a single place required the development of a world-wide distributed computing system, which allows efficient data access and makes use of all available computing resources. Thus ATLAS Collaboration embraced a computing model [6] based on the Grid paradigm and on a high degree of decentralization and sharing of computing resources. The main difference with the previous CERN-based Computin Models is that off-site facilities play a vital role to the operation of ATLAS. In particular, since 2002 a complex set of tools and distributed services, enabling the automatic distribution and processing of the large amounts of data, has been developed and deployed by ATLAS in cooperation with LHC Computing Grid (LCG) Project [16], [17] and with the middleware providers of the three large Grid infrastructures: EGEE [18] in most of Europe and the rest of the world, NorduGrid [19] in Scandinavian countries and OSG [20] in the United States. The Computing Model designed by ATLAS is a hierarchical structure organized in different *Tiers* [6]. The Tier0 facility is based at CERN and is responsible for first–pass processing and archiving of the primary Raw data and their distribution to the Tier1s. The 10 Tier1 facilities word-wide distributed have to store and guarantee a long-term access to Raw and derived data and provide all the reprocessing

activities. Each Tier1 heads up a set of Tier2 sites grouped in regional *Clouds*. Tier2s are the centres designed for the users analysis and provide all the Monte Carlo simulation capability. A scheme of ATLAS Cloud Model is represented in fig. 5. In particular, the Tier1 in Italy is located at INFN-CNAF in Bologna, while Tier2 facilities are four: Frascati, Milano, Napoli and Roma hosted by the local Universities and INFN Divisions.

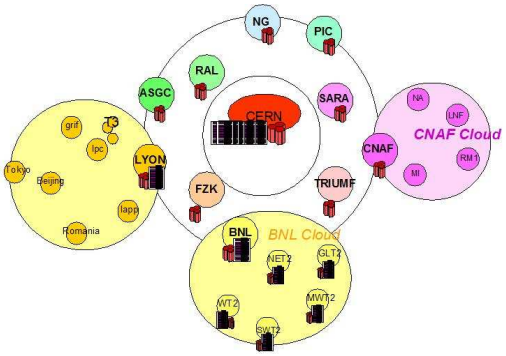


Fig. 5: ATLAS Tier Cloud Model

The main components of such system are described in the following.

1) ATLAS Virtual Organization: in order to access grid resources, ATLAS members are organized in *Atlas Virtual Organization*; the computing information and the privileges of all ATLAS members are stored in a VOMS database.

2) Distributed Data Management: Data distribution takes place through the Distributed Data Management (DDM) system [21]. Such system has two main tasks:

- to provide data replication in the collaboration sites by using the different data transfer protocols in a transparent way;
- keep tracks of data distribution in Data Catalogs.

A collection of files having similar characteristics compose a dataset; user requests of dataset replicas, called subscriptions, are provided by DDM system, which searches for data in the Catalogs and submits transfers. Information about data replicas are stored in a single database and one of the easier way to access such information is the ATLAS Dashboard portal [22]. The DDM software (DQ2) has been developed as a set of independent clients and services belonging to three categories:

- Central Services, which contain all information about content, configuration, replication and organization of data;
- Local Services, which consist of a set of agents interacting with Central Services in order to perform data transactions required;
- End User tools, which allow users to access data information, download or upload data, request subscriptions.

3) ATLAS Production System: a central system, ProdSys/Panda production system, schedules all organized data processing and simulation activities. The production System makes use of the computing resources enabled by the three different middlewares (LCG/EGEE, NDGF/Nordu-Grid and OSG/Panda) joining ATLAS VO; his workload is divided up into similar jobs, called tasks, which perform different data transformation (i.e. event generation, detector simulation, Monte Carlo digitization and reconstruction, etc.). These tasks are assigned to clouds by Panda according to a load-balancing scheme tuned to take into account workload, storage capacity and input data. Moreover, the system prevents potential inefficiencies in grids by trying each job a maximum number of times; if all the attempts fail, jobs enter in a stuck state and a await additional management operated by a team of experts.

4) ATLAS Monitoring System and User Support: This complex structure requires a continuous monitoring by all collaboration sites, as well as an efficient and monitored distribution of data and of the software needed for data processing. In particular, Production System, DDM and Central Services are monitored by a distributed team of shifters which make use of a set of quick and efficient tools providing

an overview of the overall status of the system, a fast detection of the problems and an efficient ticketing system in order to solve them in collaboration to sites managers. In addiction, support for distributed tools is performed by developers and operators using specific forums which allow both solving problems and debugging softwares.

5) ATLAS Distributed Software: ATLAS software distribution [25], is provided by ATLAS Software Installation System for LCG/EGEE [23], [26], based on the Light Job Submission Framework for Installation (LJSFi), an independent job submission framework for generic submission and job tracking in EGEE. It is responsible for access, monitor, install, test e publish information about all resources made available to ATLAS Collaboration by LCG. The core system is based on the Installation DataBase which stores the information about the resources, the jobs and the software deployment status, while the command line interface is responsible of the interaction with the Grid middleware. The software installation is performed in 3 steps. First a pilot job is sent to each site where to install the software; if site checks have been successfully executed, the actual installation process may start. The installation actions are performed in the target nodes (Worker Nodes) by the software management script, handling installations and removals of the ATLAS software distribution kit with Pacman, and for each software release installed at a site, a tag is published to the corresponding Computing Element at the end of the installation task.

6) ATLAS Distributed Analysis Tools: ATLAS Computing Model provides also a series of useful tools for distributed data analysis. There are two official tools used by ATLAS collaboration: Pathena [27], mainly used in OSG, and GANGA [28], used in EGEE and NorduGrid. The main feature of GANGA is to provide a transparent command line or graphical interface between the user and the grid infrastructure, hiding grid technicalities to users; in fact, by setting few parameters in a Python script it provides access to file catalog, computing elements, ATLAS releases needed to process data, and retrieve output data. In GANGA each job is configured by a set of functional blocks (see fig. 6), each one with a well defined purpose, in order to define:

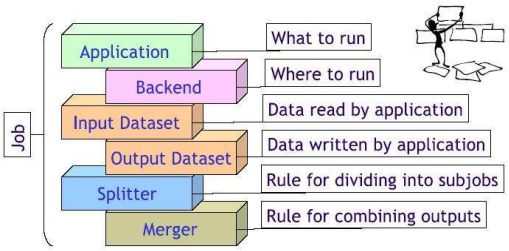


Fig. 6: GANGA functional blocks

- the application to run within the framework ATHENA or not;
- the preferred sites where to run, which may be local backends or all the other backends belonging to the three middlewares presently used by ATLAS;
- the input data needed by application, which may belong to local datasets or to DQ2;
- the output data to store;
- the task to split jobs into subjobs and to combine output.

IV. NAPOLI ATLAS TIER 2

Napoli ATLAS Tier2 started his operations as such in the first half of 2006 after many years of long experience as local computing farm. It provides resources and services to the whole ATLAS collaboration, and in particular to italian users, for Monte Carlo simulation and analysis. The main activities carried out by the Napoli and the other italian groups in the Tier2 are the Muon Triggers studies (LVL1 and Event Filter), Muon Reconstruction, RPC calibration and analysis and Supersymmetry searches. The computing resources at the end of 2008 consists of 50 Worker Nodes and 260 cores corresponding to a computing power of about 580 kSI2k, while storage resources, based on a SAN architecture, consist of 260 TB and 12 disk servers. Moreover, local services have been implemented to manage and monitor the whole

system: the Computing Element, the Storage Elements, the Storage Resource Manager DPM, and the account server HLR.

Many tests, during which dummy datasets are shipped to Tier1s and Tier2s from the Tier0 and the other Tier1s, have been performed to optimize the configuration of the system and evaluate the its performance such as:

- Functional Tests, which periodically evaluate data transfer functionalities to sites;
- Throughput Tests which verify the foreseen data transfer speed;
- CCRC08 (Combined Computing Readiness Challenge 2009), the combined test with the other LHC experiment in order to verify the capability of the common service of sustaining their contemporaneous activities.
  performances with others LHC experiments.

Moreover, other tests called Job Priorities Test are used to evaluate the correct working of fair share mechanism, which allows site manager to define which percentage of the resources can by accessed by the different user categories[1] according to ATLAS VO policies, while Distributed Analysis Tests (Hammer Tests) control the correct functionality of GANGA on the different sites.
All these tests had excellent results, according to the general trend of other Tier2s.

A. PON S.Co.P.E. Project

S.Co.P.E. [29], [30] is a project of the University "Federico II" of Napoli whose aim is to set-up a general purpose super-computing infrastructure based on Grid model and on the most recent distributed computing technologies, in order to support the scientific activities of the research groups belonging both to Napoli University and to the Universities and Research Institutes of the PON joining the GriSù Project.

[1]In ATLAS VO users may have different roles on the basis of the type of jobs they run on grid. For example, who submits central production has the 'production' role, while users who sends analysis jobs have 'none' role

On the bases of a pre-existing metropolitan network infrastructure, operating a gigabit connection between the most important research structures, S.Co.P.E. integrates the current computing and storage resources with new high performance hardware, in order to create a unique Grid platform integrated with national and international Grid infrastructures. Moreover, S.Co.P.E. project is promoting scientific development for fundamental research and technologic innovation in four research areas:

- Macrocosm and Microcosm Sciences,
- Matter and Environment Sciences,
- Life Sciences,
- Middleware,

and a large scientific community, in collaboration with INFN and other local and national research structures, is working for this purpose.

B. Napoli ATLAS Tier2 and S.Co.P.E.

ATLAS is one of the project belonging to S.Co.P.E. in the Macrocosm and Microcosm Sciences area. Its Tier2 is hosted both in the original INFN site in the Physics Department of the University and in the new S.Co.P.E. site (7). In order to ensure a fast and efficient

Fig. 7: S.Co.P.E. site.

connection of the computing resources, a 10 Gbps connection has been realized between the two sites.
Different ATLAS grid infrastructures have been implemented in order to allow ATLAS VO user to access, requesting ATLAS software installation and using GANGA on GRID-S.Co.P.E. resources. In particular, GRID MAP file has been modified in order to allow ATLAS Napoli

users to run their own jobs on a dedicated queue. ATLAS software Installation System has been readapted on the "Scope Atlas Installation System" and its correct working on S.Co.P.E. resources has been tested.

Finally, GANGA has been configured in order to use the S.Co.P.E.'s Computing Element and Resource Broker, and ATLAS jobs have been launched using this tool. A large production of simulated single muon events has been realized using the S.Co.P.E. resources. Since the interoperability between the ATLAS and the S.Co.P.E. Grid has been verified, the activities of the ATLAS experiment will exploit in future the resources of the Project accordingly to the fair share policies among the supported VOs.

V. Conclusion

The LHC Physics Program has a breakthrough discovery potential, will make an enormous number of precision measurements, and therefore is the leading component of the world High Energy Physics program from for the next 20 years. LHC will be taking data starting from next autumn over the whole next year. The ATLAS computing model had been defined in order to exploit the advantages of the GRID and the large amount of resources spread over the world. We have tested the functionality of the ATLAS software on the S.Co.P.E. and, thanks to the inter-operability, on the other PON resources.

References

[1] O.Brüning et *al.*, *"LHC Design Report Volume I: the LHC Main Ring"*, CERN-2004-003-V-1, http://cdsweb.cern.ch/record/782076, 2004.

[2] ATLAS Collaboration, *"ATLAS: technical proposal for a general-purpose pp experiment at the Large Hadron Collider at CERN"*, CERN-LHCC-94-43, http://cdsweb.cern.ch/record/290968, 1994.

[3] CMS collaboration, *"CMS technical proposal"*, CERN-LHCC-94-38, http://cdsweb.cern.ch/record/290969, 1994.

[4] LHCb collaboration, *"LHCb technical proposal"*, CERN-LHCC-98-004, http://cdsweb.cern.ch/record/622031, 1998.

[5] ALICE collaboration, *"ALICE: Technical proposal for a Large Ion collider Experiment at the CERN LHC"*, CERN-LHCC-95-71, http://cdsweb.cern.ch/record/293391, 1995.

[6] ATLAS Computing Group, *"ATLAS computing: Tecnichal Design Report"*, CERN-LHCC-2005-022, http://cdsweb.cern.ch/record/837738, 2005.

[7] S. L Glashow, *"Partial-Symmetries of Weak Interactions"*, Nucl. Phys. **22**:579, 1961.

[8] A. Salam, *"Weak and Electromagnetic Interactions"*, in *"Elementary Particle Theory: Groups and Analycity"*, pag.367, ed. Svartholm; Almquist and Wiksell, Stocholm, 1968.

[9] S. Weinberg,*"A Model of Leptons"*, Phys. Rev. Lett.**19**: 1264, 1967.

[10] P. W. Higgs, *"Spontaneous Symmetry Breakdown without Massless Bosons"*, Phys. Rev. Lett. **145**:1156, 1966.

[11] ALEPH Collaboration, DELPHI Collaboration, L3 Collaboration, OPAL Collaboration and The LEP Working Group for Higgs Boson Searches, *"Search for the Standard Model Higgs Boson at LEP"*, Phys. Lett. B **565**:61-75, 2003.

[12] F. Giannotti, *"Searches for supersymmetry at high-energy colliders: the past, the present and the future"*, New Journal of Physics **4**: 63.1-63.25, 2002.

[13] S. P. Martin, *"A Supersymmetry Primer"*, hep-ph/9709536, 1999.

[14] http://cern.ch/proj-gaudi/ .

[15] http://physics.bu.edu/pacman/ .

[16] I.Bird et *al.*, *"LHC Computing Grid: Tecnichal Design Report"*, CERN-LHCC-2005-024, http://lcg.web.cern.ch/LCG/tdr, 2005.

[17] L. Perini, *"Overview of the contributions of the LHC experiments in INFN GRID for bringing the Grid to production quality"* , Proceedings of the same Workshop.

[18] E.Laure and B. Jones, *"Enabling Grids for e-Science: The EGEE Project"*, EGEE-PUB-2009-001, http://cdsweb.cern.ch/record/1128647, 2008.

[19] P.Eerola et *al.*, *"The NorduGrid architecture and tools"*, in Proceedings of CHEP 2004, eds. A.Aimar, J.Harvey and N.Knoors, CERN-2005-002,Vol.2, 2005.

[20] R.Pordes, *"The Open Science Grid (OSG)"* , in Computing in High Energy Physics and Nuclear Physics, http://cdsweb.cern.ch/record/865745, 2004.

[21] http://dashboard.cern.ch/atlas/.

[22] http://dashb-build.cern.ch/atlas/.

[23] https://atlas-install.roma1.infn.it/atlas_install/.

[24] E.Obreshkov et *al.*,*"Organization and Management of ATLAS Offline Software Releases"* , in Nucl. Instrum. Methods Phys. Res. **A**, 2006.

[25] S.Gonzales et *al.*,*"Distributed analysis job with the ATLAS production system"* , in IEEE Trans. Nucl. Sci. **53**, 2006.

[26] A. De Salvo et *al.*, *"The ATLAS Software Installation System for LCG/EGEE"* , Journal of Physics: Conference Series **119** 052013, 2008.

[27] https://twiki.cern.ch/twiki/bin/view/Atlas/PandaTools.

[28] http://ganga.web.cern.ch/ganga/ .

[29] L. Merola, *"The S.Co.P.E. Project"* , Proceedings of the same Workshop.

[30] http://www.scope.unina.it/progetto/default.aspx.

The Astronomy and Astrophysics Cluster in EGEE-III

C. Vuerli[1,2], G. Taffoni[3], F. Pasian[1,2], M. Sponza[1], U. Becciani[4] and S. Cassisi[5]

[1]*INAF - Astronomical Observatory of Trieste, Trieste, Italy*
[2]*INAF - Information Systems Unit, Trieste, Italy*
[3]*Sincrotrone Trieste SCpA, Trieste, Italy*
[4]*INAF - Astrophysical Observatory of Catania, Catania, Italy*
[5]*INAF - Astronomical Observatory of Teramo, Teramo, Italy*

Abstract — **The A&A (Astronomy and Astrophysics) cluster in EGEE has been set up in January 2007. Its creation was strongly encouraged and supported by EGEE NA4 aiming at finding new scientific communities and applications for EGEE. Astronomy and Astrophysics have been immediately recognized as key disciplines given the success of two pilot astrophysical applications in EGEE since 2004, Planck and MAGIC. Astronomical applications, in fact, with their challenging requests in terms of computing power, storage and other non-standard services represent excellent test-beds for EGEE and its Grid infrastructure. The creation of the A&A cluster was very important for two main reasons: a) new use-cases were proposed for EGEE and new astrophysical applications were ported on its infrastructure; b) sparse astronomical communities already present in EGEE but forced to ask support to other scientific communities found in the new cluster the necessary support (in terms of resources, expertise and training) to carry out their activities with the Grid in the best possible way. The creation of the new cluster, in fact, was accompanied by the creation of a dedicated A&A VO mainly thought for Grid beginners who need to be trained in approaching and using the new technology. This general-purpose VO joined already existing production VOs tailored on the needs of specific applications like MAGIC and Planck. In EGEE-III some funds have been allocated to support disciplinary clusters in their activities; in this way it was possible to hire some man-power mainly dedicated to the porting of applications and to training activities. Many astrophysical Institutes and research groups are now part of the cluster. Key subtasks of the cluster work plan are: a) go on with the applications porting; b) prepare the A&A community in view of the transition to EGI; c) train new groups in learning and using the Grid technology; d) go on in the effort of making the Grid and the VObs (Virtual Observatory) fully interoperable.**

Index Terms — **EGEE, Disciplinary Cluster, Astronomy, Astrophysics, Applications Porting, Training, Dissemination, EGI, Virtual Observatory.**

I. INTRODUCTION

DISCIPLINARY clusters are an important component of the EGEE-III project [1]. Their coordination is in charge of EGEE/NA4 network activity whose main goal is to expand and support the community of users. The A&A cluster [2] is in EGEE since January 2007 and joined other important disciplinary clusters, namely Computational Chemistry, Earth Science, Fusion, Grid Observatory, High-Energy Physics and Life Sciences. The creation of the cluster for Astronomy and Astrophysics represented an important step in the direction of widening and consolidating at the same time the presence of the astronomical community in the biggest and most important European Grid infrastructure currently in place. With the third edition of EGEE a minimum amount of funds were allocated to disciplinary clusters, although far to significantly support the necessary effort requested for a rapid growth of the user communities; they made possible

however to hire some man-power mainly dedicated to the application porting activities. Despite the lack of funds, the astrophysical community continued to grow with many groups engaged in the process of porting in Grid their applications, although with no funds available. A significant pool of applications now run on the European infrastructure and the community is mature to move to EGI [3], the stable European Grid Initiative currently under construction and ready to enter in its operative phase at the end of EGEE-III (May 2010).
This paper illustrates the evolution of the A&A cluster in EGEE, its current structure and the necessary steps to be undertaken in the next months as requested by the migration process.

A. Motivation

The Astronomy and Astrophysics (A&A) community that could join the EGEE infrastructure is potentially wide. This community is sparse all around Europe and counts a remarkable number of users and institutes. The creation of an A&A cluster in EGEE, therefore, means first of all more users that join the EGEE infrastructure bringing a rich variety of applications that run on the Grid and make use of available shared resources. Many A&A applications are quite challenging from the point of view of resources, both in terms of processing capabilities and data storage, so they are an excellent test-bed for Grid infrastructures. Until the activation of the A&A cluster each Institute has frequently provided from its own to the procurement of adequate resources to make applications able to run on local computing facilities (mainframes and/or local clusters of PCs). Quite often Institutes agreed with some supercomputing centres the usage of the necessary resources. In Italy many universities and also INAF (the National Institute of Astrophysics) have active conventions with CINECA, one of the most important national supercomputing centres. As A&A applications grow in complexity, however, single institutes cannot afford application demands on their own. This is true also for supercomputing centres that, in turn, tend to establish partnerships in the framework of trans-national projects and use the Grid to share their own resources.

A&A applications, moreover, do not only need computing power and conventional data storage but often require non-conventional resources as well, namely access to astronomical databases as well as remote monitoring/control of astrophysical instrumentation. A so enhanced infrastructure for EGEE opens new opportunities not only for A&A but also for other scientific communities.

For this reasons many A&A groups are engaged in the process of modifying some components of the Grid middleware to add these new functionalities being EGEE the main target infrastructure.

Finally, many A&A applications well suit the Grid but often their developers do not know how to approach the gridification process. Within the A&A cluster the network of users can exchange information and expertise in successfully approaching this task. Man-power support is mainly expected by EGEE in this case.

In conclusion, with the A&A cluster, EGEE includes a wide and motivated community of users, a rich variety of applications running on its Grid infrastructure and a set of new functionalities that could make EGEE attractive also for other communities of users. In return A&A users find in EGEE a pool of resources, tools and services able to improve performances of their applications and a suitable environment where their scientific and technological collaborations can be further developed and extended.

B. The A&A Cluster in EGEE

The first astrophysical applications approved by EGAAP (EGEE Generic Applications Advisory Panel) in 2004 were Planck [4] and MAGIC [5]. For the communities of users contributing to these applications, ad-hoc VOs

(Virtual Organizations) were created. Planck and MAGIC can be considered the first pilot applications as they demonstrated that the usage of the Grid can be successful for a large fraction of astrophysical applications. They also proved that these applications are challenging for Grid infrastructures in terms of number and type of requested resources. For this reason the astrophysical community was encouraged to create its own cluster in EGEE. It was set up in January 2007 under the coordination of INAF that in 2004 proposed Planck, one of the two pilot applications. Soon a growing number of research groups and Institutes joined the cluster and nowadays important communities of astronomers are part of the cluster.

Initially all groups operated within the cluster with no funds available. In summer 2007 the proposal for the third edition of EGEE (EGEE-III) was submitted to the European Commission and a fraction of the allocated funds was used to support disciplinary clusters. The amount of funds assigned to the A&A cluster was largely insufficient to support all research groups; it was possible to allocate funds for only five of the eighteen partners of the cluster. Despite this, activities aimed at porting new applications in Grid and at developing tools and services according to application demands continued. Many groups, although unfunded, brought their contributions to the cluster and this is the basis of its success and the good level of achieved maturity. At present the cluster is composed by communities of users and developers of nine European countries.

1) Applications and Services

Many contributions in terms of applications come from the different research groups currently operating within the cluster.

The MAGIC application is now under development at PIC (Port d'Informacio Cientifica) in Barcelona with contributions coming also from UCM (Universidad Complutense de Madrid). MAGIC is a telescope that observes gamma-rays in the energy-range between 30 GeV up to 1 TeV and the purpose of the application is to bring in Grid the Monte Carlo code used to simulate the production of events.

Two groups currently work on Planck simulations; a) at INAF Trieste (where the Data Processing Centre of the Low Frequency Instrument is hosted) the gridification process of simulation code is going on; b) the group operating at IFCA (Instituto de Fisica de Cantabria), Santander, mainly deals with extragalactic compact source detection and with the analysis of the Gaussianity of Planck data.

INAF moreover is active in the cluster with two important additional applications: the first one deals with the problem of galaxy formation at high-redshift and more specifically with the problem of quantifying the Black Holes feedback in the evolution of the visible Universe (Observatory of Catania); the second application deals with the gridification of FRANEC, the simulation code feeding BaSTI [6], a database containing stellar evolution tracks, isochrones and codes to study the properties of resolved and unresolved stellar populations with an arbitrary Star Formation History (Observatory of Teramo).

At the Slovak Academy of Sciences several applications are under development; they range from SWIFT, an application aiming at researching the structure of the reservoirs of small bodies in the outer region of the Solar System, to other minor applications. Specific areas of interest in which II-SAS is active include: a) reliable execution of parametric simulations; b) parallel simulations using MPI; c) interactive applications; d) data management and user friendly interfaces to interact with Grid technologies (portals, visualisation tools, workflows).

The group operating at RUG (Rijksuniversiteit Groningen) deals with LOFAR [7], currently the largest radio

telescope in the world observing at low frequencies (1 - 250 MHz). The purpose of the related application is the gridification of the code developed to process data collected by the radio telescope.

All the partners listed above operate in the cluster with a very minimum amount of allocated funds coming from EGEE-III.

Other unfunded research groups actively contribute to the cluster with interesting applications. The French community is one of the most proactive in this sense with several Astronomical Observatories involved (Paris Meudon, Lyon, Strasbourg, Grenoble and the Observatoire de la Côte d'Azur). The most important applications under development by this community relate Horizon and the Herschel-ALMA project. Other activities include the interactions between the Grid and the VObs, image processing and workflows, stellar atmospheres simulations, fluids mechanics and astrophysical plasmas until minor projects related to Celestial Mechanics and High Energy Astrophysics.

It is also worth of mention the activity in progress in Germany with a tight synergy and a fruitful collaboration between Astrogrid-D, the German Grid infrastructure for Astrophysics and GAVO, the German Astrophysical Virtual Observatory and at CESNET (Prague, Czech Republic) where code related with the Pierre Auger Cosmic Ray Observatory [8] is under gridification.

2) Virtual Organizations

Virtual Organizations in EGEE are of two types: VOs set up for the needs of specific projects (Planck and MAGIC are two examples in the astrophysical domain) or more generic VOs for the benefit of wider user communities. Apart from Planck and MAGIC, a generic VO has been created for the astrophysical community at the same time of the activation of the cluster. This kind of arrangement of VOs should allow people who approach the Grid for the first time to find and allocate resources in generic VOs. Once the user acquires the necessary skill and gets involved in some specific project, he/she has access to resources available in specific VOs.

All VOs currently active for astronomers are supported by the most important VOs in EGEE. This means that astronomers can use resources of these VOs or at least a fraction thereof.

The generic VO for the astrophysical cluster is currently managed at INAF Trieste as well as the Planck VO. The MAGIC VO is managed in Barcelona.

C. Cluster Structure

The astrophysical cluster is currently coordinated by INAF Trieste as requested by people of EGEE/NA4 at the time of its creation. The coordinator is member of the EGEE/NA4 SC (Steering Committee) being the main goal of NA4 [9] the expansion of the user community and its support.

The cluster is mainly composed of participating Institutions with their research groups each of them contributing to the cluster with various tasks (management, porting of applications, development of tools and services, training and dissemination, etc.). Tasks in which research groups are involved have been identified at the time of definition of the cluster work plan.

1) Partner Institutes

Eighteen partner Institutes form currently the cluster. They are reported in table below where the first five Institutes are those that got funds in EGEE-III.

IFAE/PIC	ES	Institut de Física d'Altes Energies/Port d'Informaciò Cientifica
IFCA	ES	Instituto de Física de Cantabria
II-SAS	SK	Institute of Informatics Slovak Academy of Sciences
INAF	IT	Istituto Nazionale di Astrofisica

RUG	NL	Rijksuniversiteit Groningen
AIP	DE	Astrophysical Institute Potsdam
ARI	DE	Astronomisches Rechen-Institut, Heidelberg University
ESA/ESAC	ES	European Space Agency European Space Astronomy Centre
FZK	DE	Forschungszentrum Karlsruhe
CESNET	CZ	Czech Republic's National Research and Education Network (NREN)
OBSG	FR	Observatoire de Grenoble
OBSL	FR	Observatoire de Lyon
OBSPM	FR	Observatoire de Paris Meudon
OBSS/CDS	FR	Observatoire de Strasbourg Centre de Données Astronomiques
OCA	FR	Observatoire Côte d'Azur
ROE	UK	Royal Observatory Edinburgh
UCAM	UK	Institute of Astronomy University of Cambridge
UIBK	AT	University of Innsbruck

2) Work Plan

The work plan was defined at the beginning of the EGEE-III project as requested by the NA4 SC. The work plan of the A&A cluster is composed of five subtasks as reported in the table below

AA1	Management
AA2	Migration to EGI
AA3	Applications
AA4	Training and Dissemination
AA5	Interactions with EuroVO

The subtask AA1 provides to: a) Periodically update the set of requirements of the astronomical applications; b) Select good demonstrators among A&A applications already ported to the grid; c) Coordinate all astronomy-related VOs created in EGEE; d) Manage the generic EGEE VO created for the A&A cluster.

Subtask AA2 include the following activities: a) collect information from A&A partners about their NGIs (National Grid Initiatives); b) promote the utilization of the Grid technology by all potential astronomical users in Europe; c) Interact with NGIs and with EGI. Further details about the migration to EGI can be found in subsection D.1).

Subtask AA3 is the core of the cluster as it deals with the porting of applications and the development of tools and services to fulfil the applications requirements. All Institutes participate to this subtask. Applications currently under the gridification process are those discussed in section B.1).

Subtask AA4 deals with training and dissemination aspects. Only funded Institutes participate to this subtasks and related activities can be summarized as follows: a) collect information on training events provided by NA3 and publicize it within the A&A community; b) encourage A&A developers to attend training events of interest for their activities; c) collect training requests coming from A&A community and pass them on to the cluster manager who in turn will route them to NA3; d) organize internal training events once they are approved by the cluster management and in coordination with the NA4 support teams and with NA3; e) publicize on the cluster TWiki web site [2] any dissemination material produced by groups of developers and notify NA2; f) analyze dissemination material produced by NA2 and link the most relevant through the cluster TWiki web site; g) evaluate in coordination with the cluster management and with NA2 any initiatives to foster the adoption of Grid technology and of the EGEE Grid infrastructure within the astronomical community.

The goal of subtask AA5 is to achieve a tight interoperability between the Grid and the EuroVO (European Virtual Observatory). This topic is discussed in detail in subsection D.2).

Each subtask has associated a number of

parameters (subtask metrics). Values associated to these parameters must be verified at every quarter period of the project and be reported in the quarterly reports. They allow to gauge the progress in achieving the goals of each subtask of the work plan.

D. Future Evolution

The current year (2009) is of crucial importance for the future evolution of Grid infrastructures in Europe. The process aimed at creating a stable Grid infrastructure, in fact, is already in progress. This process will terminate with the creation of a central entity named EGI.org (EGI Organization) and with the creation of NGIs (National Grid Initiatives) in all countries that accepted to adhere to the European Grid Initiative. EGI indeed is made of NGIs; a centralized organization will guarantee the necessary coordination between the various NGIs but the core activities of the European Grid will be carried out by the various National Grid Initiatives. The process of constitution of the European Grid is already in progress. The EGI_DS (EGI Design Study) was activated and will be in place until the end of 2009. Its role is to formulate proposals for the constitution of EGI. A fundamental step has already been carried out: EGI_DS produced the EGI Blueprint that has been approved in January 2009. The structure of EGI, therefore, is now defined: the constitution of EGI (their governing bodies and central offices) as well as the creation of NGIs can now take place. The process of constitution of EGI and of the NGIs (at least those of the countries that currently play a leading role for what concerns the European Grid) is requested to complete by the end of 2009. The first months of 2010 will be spent to verify that the new constituted entities are fully functioning and to apply corrective actions if necessary. EGI and the constituting NGIs will enter into operations on May 1st 2010 when the EGEE-III project terminates.

1) Migration to EGI

Given this important reorganization of the European Grid Infrastructures, scientific communities now in EGEE as disciplinary clusters are requested to start the preparation of the migration to EGI. It is worth to note here that EGI cannot be simply considered a new edition of the EGEE project. Rules within EGI will be quite different with respect those that today regulate the participation to EGEE. According to the most probable scenario, in the EGI era disciplinary clusters should organize themselves in the so called SSCs (Specialized Support Centers). Within EGI, SSCs are entities highly independent from the point of view of structure and internal organization. Probably they will be requested to procure funds from their own by submitting a specific project to the European Commission, although for the time being this funding mechanism for SSCs was not confirmed yet.

The A&A cluster therefore must activate itself in the next months to move to EGI according to rules that will be defined.

A first very preliminary proposal for the A&A SSC (especially for what concerns its internal organization) is already in place; starting from this first proposal members of the A&A cluster must take a decision in the next months about how to create their SSC. It is foreseen however that NA4 will try to drive this process through a coordinated action among all disciplinary clusters.

The preparation of a project to be submitted to the EC is certainly the most critical step in this process if this funding mechanism will be confirmed in the next months. It is requested therefore a strong coordination within the cluster in the next months to afford the necessary steps towards EGI in the best possible way.

2) Interactions with EuroVO

The process of organizing the European astrophysical community in a SSC of EGI must involve all research groups of the cluster, both

funded and unfunded. During the next months one of the most challenging tasks that will require a strong coordination action and a massive dissemination activity is the expansion of the A&A community by involving new research groups that have not participated to the cluster so far.

One of the most important issues that should be afforded in view of the migration to EGI is the improvement of the synergy between the EGEE A&A cluster and EuroVO, the European component of the Virtual Observatory. This should happen at two different levels: a) more interactions and a stronger level of collaboration between groups operating within EGEE and EuroVO; b) an improved level of interoperability between the two technologies and hopefully the construction of a fully integrated environment for the benefit of the whole community of users. Very often working groups develop tools and services in the two environments with no interactions at all. According to the past experience and having in mind the current interoperability issues, the level of communication between groups operating for the Grid and for the VObs should definitely improve in the future.

II. Conclusion

The astrophysical cluster in EGEE has achieved now a good level of maturity. A noteworthy portfolio of advanced applications, tools and services are the result of the contributions brought to the cluster by the various working groups, both funded and unfunded.

In the next months a rather challenging task must be tackled by the astronomical community: the transition of the current cluster to EGI as a SSC. This critical transition must be carefully driven; it requires to be well coordinated and a good level of collaboration among the various research groups that are requested to give their contributions for the success of the migration.

The transition to EGI is also a good occasion for a further expansion of the astrophysical community currently engaged in EGEE with the involvement of new Institutes and research groups in the exercise of using the Grid for the astrophysical research. The first groups that should be involved in this process however are those that now operate and develop code for the European Grid Observatory.

Acknowledgments

The authors would like to thank all colleagues that with their work made possible the success of the astrophysical cluster in EGEE. A special thank to Charles Loomis (LAL, Paris) and to Evangelios Floros (GRNET, Athens) for their effective coordination of NA4 and of its SC.

References

[1] http://www.eu-egee.org/
[2] https://twiki.cern.ch/twiki/bin/view/EGEE/AstronomyAndAstrophysics
[3] http://web.eu-egi.eu/
[4] http://www.rssd.esa.int/index.php?project=planck
[5] http://wwwmagic.mppmu.mpg.de/
[6] http://albione.oa-teramo.inaf.it/
[7] http://www.lofar.org/
[8] http://www.auger.org/
[9] https://twiki.cern.ch/twiki/bin/view/EGEE/NA4
[10] Authors, "Title," Name conference or book, Editors, year, pages.

C. Vuerli is a Research Astronomer: He took his degree in Computer Science in 1985. Currently he's involved and coordinates all Grid-related projects at INAF-SI and INAF-Astronomical Observatory of Trieste. He is responsible for managing activities finalized to make the Grid a suitable tool for astrophysical applications. Since August 2005 he is staff of SI, the INAF structure having in charge the national coordination of all services related with data processing, data archiving and network infrastructure. Since January 2007 he coordinates the Astrophysical cluster (A&A cluster) created within the EGEE project.

G. Taffoni He has got PhD in 2001 at SISSA and a post-doc at Astronomical Observatory of Trieste on HPC and data reduction/analysis. He has got great expertise for what concerns HPC, system management and the Grid technology acquiring deep knowledge of various implementations of the Grid middleware. He also is the technical reference person within INAF-SI at Trieste for what concerns the interoperability between the Grid and local clusters (Worker Node) and the porting of applications on the Grid. He is referee for international journals and author or co-author of 24 publications in main international journals, conference proceedings, technical papers.

F. Pasian is Full astronomer. He's the Director of INAF-SI (INAF-Information Systems) for the national coordination of all services related with data processing, data archiving and network infrastructure used in astrophysics. He has been active in the field of astronomical image processing, data archiving and data processing. He has led many projects both at national and international level such as the archive of the Hubble Space Telescope (HST), the archive at the Galileo National Telescope (TNG), the Data Processing Centre of the Low Frequency Instrument of the Planck mission of ESA. He's author/co-author of about 300 publications and technical notes.

M. Sponza is graduated in Computer Engineering from the University of Trieste. He worked for several years as a system administrator and on application development and integration on Microsoft Windows platform at Liksystem srl. After that he was UNESCO/IAEA Consultant at the Abdus Salam International Center for Theoretical Physics ICTP (http://www.ictp.it), involved in egrid project (http://www.egrid.it) to build the Italian Pilot National GRID Computing facility for research in Economics and Finance. Currently he is a collaborator at INAF-OATs (http://www.oats.inaf.it) in EU funded EGEE-III project.

U. Becciani was born in Messina in 1959, Italy. Permanent position: Astronomer Researcher (computing science field), at the INAF - Astrophysical Observatory of Catania. He is author of more than 25 specialistic papers published on international journals and more than 70 proceedings of international conferences. He is also referee of Journal of Computer Physics Communication and Journal of Computational Physics, and referee for National Science Foundation to review proposals in computational physics submitted to the Physics at the Information Frontier (PIF) Program. In 1999 - 2008, he has been PI and CoPI of several research projects on different astrophysical topics, data exploration, grid computing, virtual observatory and advanced studies in supercomputing. The main project are: Cosmo.Lab for scientific visualization, funded by EC with 378 KEuro, VOTECH (EuroVO) responsible for VisIVO tool, TriGrid VL for grid computing (infrastructure) funded by Sicilian Regional Government P.O.R. with 3,500 Euros, PI2S2 (Cometa consortium) for research on grid computing and infrastructure funded by Italian Ministry for Research P.O.N. 11,400 KEuros.

S. Cassisi got his degree in Physics in 1991 at University of Pisa. Grant of Ministery for Scientific Research of the French Government since 1993 to 1994, spent at the Astronomical Observatory of Meudon-Paris. PhD fellowship in Physics at the University of L'Aquila since 1995 to 1997. Staff researcher at the INAF - Astronomical Observatory of "Collurania" - Teramo since 1998. Current position: Associate professor at the INAF - Teramo Observatory. Tutor of several degree and PhD thesis in Physics and Astronomy. Since 2006, on behalf of INAF, he is one of the members of the TEG at EUROVO. The main research fields are theoretical stellar evolution and its application to the study of both galactic and extra-galactic stellar populations. In this field, about 190 scientific papers have been published so far (about 106 papers are referred publications) and a monography for students and researchers.

Study of the Effects of Nuclear Motion on High Harmonic Generation in Simple Molecules

G. Castiglia[1], P. P. Corso[1], R. Daniele[1], E. Fiordilino[1], F. Morales[2] and G. Orlando[1]
[1]*Dipartimento di Scienze Fisiche ed Astronomiche, Università degli Studi di Palermo, via Archirafi 36, 90123 Palermo, Italy and CNISM.*
[2]*Dipartimento di Fisica e Tecnologie Relative, Università degli Studi di Palermo, Edif. 18, Viale delle Scienze, Parco d'Orleans, 90128 Palermo, Italy and CNISM.*

***Abstract*—The dynamics of hydrogen-like molecules and ions, interacting with strong laser fields, is investigated beyond the frozen nuclei approximation. All the results presented in this paper have been obtained numerically solving the time-dependent Schroedinger equation (TDSE) of the molecular systems. Our numerical simulations show that the nuclear motion introduces interesting features in the emitted spectrum. In particular we observe the presence of sidebands in the emitted spectra around the usual odd harmonics, i.e. additional regular lines whose separation is strictly related to the vibrational frequency of nuclear motion.**

***Index Terms*—Atoms and molecules in strong laser fields, high-harmonic generation (HHG), molecular vibrations, time-dependent Schroedinger equation (TDSE).**

I. Introduction

Atoms and molecules driven by a strong laser field emit electromagnetic energy whose spectrum consists of well defined lines at frequencies that are multiples (up to 200 or more) of the laser frequency (HHG: high-harmonic generation) [1]–[3]. The envelope of these spectra shows a rapid decrease of the first emitted lines followed by a large plateau of roughly constant amplitude and finally by a rather abrupt cut-off. This process arises from the non linear response of the electron wave function to intense laser fields: an electron is driven by the laser field away from its parent atom or molecule, propagates in the field and returns to the parent ion with emission of high-energy photons [4].

The emitted radiation can be used in the production of UV or XUV pulses and to probe the electron dynamics; it carries information on the emitting source allowing a tomographic image of atoms and molecules [5]. The atomic spectra present features largely independent of the atomic species; on the contrary the large number of molecular degrees of freedom makes the radiation scattered from molecules strongly dependent on the molecular species. The molecular harmonic spectra thus carry information on the molecular structure, namely on the dynamics of both electrons and nuclei [6].

The electron dynamic in laser driven atoms and molecules has been extensively investigated. In this paper we show the possibility of monitoring the nuclear vibration of a molecule. In particular, we investigate the response of a diatomic molecule to a monochromatic field, whose intensity is low enough to prevent a high rate of ionization and dissociation of the molecule, but strong enough to leave the nuclei in a slow oscillatory motion.

II. Theory and model

In our investigation, we have studied the dynamics of H_2, D_2 molecules and of H_2^+, D_2^+

molecular ions in the presence of an external laser field. The TDSE of a such systems is:

$$i\hbar\frac{\partial}{\partial t}\psi(\mathbf{R},\mathbf{r},t) = H\psi(\mathbf{R},\mathbf{r},t) \qquad (1)$$

where $\mathbf{R}$ and $\mathbf{r}$ indicate the totality of nuclear and electronic coordinates respectively (for H_2 and D_2 $\rightarrow$ ($\mathbf{R}$, $\mathbf{r}$) $\equiv$ ($\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{r}_1$, $\mathbf{r}_2$); for H_2^+ and D_2^+ $\rightarrow$ ($\mathbf{R}$, $\mathbf{r}$) $\equiv$ ($\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{r}_1$)). H is the Hamiltonian of the system given by:

$$H(t) = -\frac{\hbar^2}{2M}\sum_i \nabla^{\mathbf{2}}_{\mathbf{R_i}} - \frac{\hbar^2}{2m}\sum_i \nabla^{\mathbf{2}}_{\mathbf{r_i}} + \\ +U(\mathbf{R},\mathbf{r}) + W(\mathbf{R},\mathbf{r},t) \qquad (2)$$

with M and m nuclear and electronic mass respectively. In the last equation the terms

$$\nabla^{\mathbf{2}}_{\mathbf{R_i}} \equiv \frac{\partial^2}{\partial R_{i_x}} + \frac{\partial^2}{\partial R_{i_y}} + \frac{\partial^2}{\partial R_{i_z}}$$

$$\nabla^{\mathbf{2}}_{\mathbf{r_i}} \equiv \frac{\partial^2}{\partial r_{i_x}} + \frac{\partial^2}{\partial r_{i_y}} + \frac{\partial^2}{\partial r_{i_z}}$$

are the kinetic operator of the nuclear and electronic coordinates, $U(\mathbf{R},\mathbf{r})$ is the electrostatic interaction energy among the nuclei and the electrons, while

$$W(\mathbf{R},\mathbf{r},t) = -e\left(\sum_i \mathbf{R_i} - \sum_i \mathbf{r_i}\right)\cdot\mathbf{E}(t)$$

is the interaction energy between the charged particles and the external classical field $\mathbf{E}(t)$ in the dipole approximation.

Eq. (1) cannot be solved analytically, especially in the strong field regime. It is thus necessary to adopt a numerical approach. Furthermore, since eq. (1) involves at least 9 degrees of freedom, it is necessary to introduce the hypotheses of reduced dimensionality, i.e. to investigate the system by one-dimensional models treated beyond the fixed nuclei approximation. In particular, for the molecular ions (H_2^+ and D_2^+) we have considered two models:

- in the first model, called *semi-classical*, the electron is treated as a quantum particle and the nuclei are considered as classical particles whose dynamics can be described by means of Newton laws [7]–[9]. This is justified by the fact that the nuclear mass M is much larger than the mass m of the electron;
- in the second model, called *full-quantum*, all the particles are treated as quantum particles.

For the H_2 and D_2 molecules we have solved their dynamics by means of only the "semi-classical model".

By using the "semi-classical model", the temporal evolution of the H_2^+ or D_2^+ molecular ion in the external laser field is evaluated numerically by solving simultaneously the TDSE for the electron and the Newton equations for the nuclei respectively. This means that the TDSE for the electron wave function ψ is given by

$$i\hbar\frac{\partial}{\partial t}\psi(X_1,X_2;x,t) = \\ = H_e\psi(X_1,X_2;x,t) \qquad (3)$$

where x identifies the position of the electron, X_1 and X_2 are the nuclear coordinates treated as external parameters and

$$H_e(t) = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + \\ +U(X_1,X_2;x) + exE(t) \qquad (4)$$

is the electron Hamiltonian. $U(X_1,X_2;x)$ is the electron-nuclei interaction potential energy assumed of the soft-core form

$$U(X_1,X_2;x) = \\ = -\frac{e^2}{\sqrt{A+[x-X_1]^2}} + \\ -\frac{e^2}{\sqrt{A+[x-X_2]^2}} \qquad (5)$$

with A an appropriate screening parameter.

The Newton equation for the i-th nucleus is:

$$M\frac{d^2}{dt^2}X_i(t) = \\ = F_{ie}(t) + F_{ij}(t) + F_{iL}(t) \\ i \neq j = 1,2 \qquad (6)$$

where

$$F_{ie}(t) = = -\int \frac{e^2[x-X_i]|\psi(X_1,X_2;x,t)|^2}{\{A+[x-X_i]^2\}^{3/2}}dx \quad (7)$$
$$i = 1,2$$

is the force exerted by the electron cloud,

$$F_{ij}(t) = \frac{e^2[X_i - X_j]}{\{A_N + [X_i - X_j]^2\}^{3/2}} \quad (8)$$
$$i,j = 1,2, \quad i \neq j$$

is the nucleus-nucleus interaction force and

$$F_{iL}(t) = eX_iE(t) \quad (9)$$

represents the force acted upon the nucleus by the laser field.

Since our investigation is mainly focused on the electromagnetic radiation emitted by the system, we have evaluated the emitted power by means of the Larmor formula

$$P(t) = \frac{4e^2}{3c^3}|a(t)|^2 \quad (10)$$

where $a(t)$ is the quantum averaged acceleration of the electron obtained by the Ehrenfest theorem, while the spectrum of the emitted radiation is calculated by means of

$$P(\omega) = \frac{4e^2}{3c^3}|\tilde{a}(\omega)|^2 \quad (11)$$

with $\tilde{a}(\omega)$ the Fourier transform of $a(t)$.

The extension of this model to the case of H_2 and D_2 molecules is immediate: in the equations from (3) to (11), where necessary, we must consider the contribution of the second electron as well.

In the second model, the *full-quantum* model, that we have adopted only for the molecular ions, the one-dimensional TDSE of the system is

$$i\hbar\frac{\partial}{\partial t}\psi(R,x,t) = H\psi(R,x,t) \quad (12)$$

where we have assumed that is possible to describe the dynamic of the system by considering two degrees of freedom: $R = |X_1 - X_2|$, the internuclear distance, and x, the electron position.

With this assumption, the Hamiltonian H of the system is given by

$$H(t) = -\frac{\hbar^2}{2\mu}\frac{\partial^2}{\partial R^2} - \frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + \quad (13)$$
$$+U(R,x) + W(R,x,t)$$

where $\mu = 0.5M$ is the reduced mass of the two nuclei,

$$U(R,x) = = -\frac{e^2}{\sqrt{A_{Ne}+[x-R/2]^2}} + \quad (14)$$
$$-\frac{e^2}{\sqrt{A_{Ne}+[x+R/2]^2}} + \quad$$
$$+\frac{e^2}{\sqrt{A_{NN}+R^2}}$$

the electron-nuclei and nucleus-nucleus electrostatic interaction energy and

$$W(R,x,t) = e(x-R)E(t) \quad (15)$$

the interaction energy with the laser field. In equation (14) A_{Ne} and A_{NN} are the screening parameters for electron-nuclei and nucleus-nucleus interactions respectively. Regarding the emitted spectra by the system we have considered, as in the *semi-classical* model, the Fourier transform of the acceleration.

III. Results

In what follow we present the results for the molecular ions by using both *semi-classical* and *full-quantum* model. In our simulations, that we have performed by using a code based on the *split-operator* method, we propagate the electron wave function starting from the ground state of the system (for more details on how calculate it, see [10]). The spatial box is $512a_0$ wide (a_0 is the Bohr radius) with a spatial grid step $\Delta x = 0.25a_0$; to prevent reflections, an absorbing mask is inserted at the boundary. The laser pulse, with peak intensity $I_0 = 10^{14}$ Wcm^{-2} and wavelength $\lambda = 780$ nm, has a trapezoidal shape with 8 optical cycles (oc) of turn-on and turn-off and constant intensity for 240 oc. Moreover, the laser intensity is not able to induce dissociations of the system but

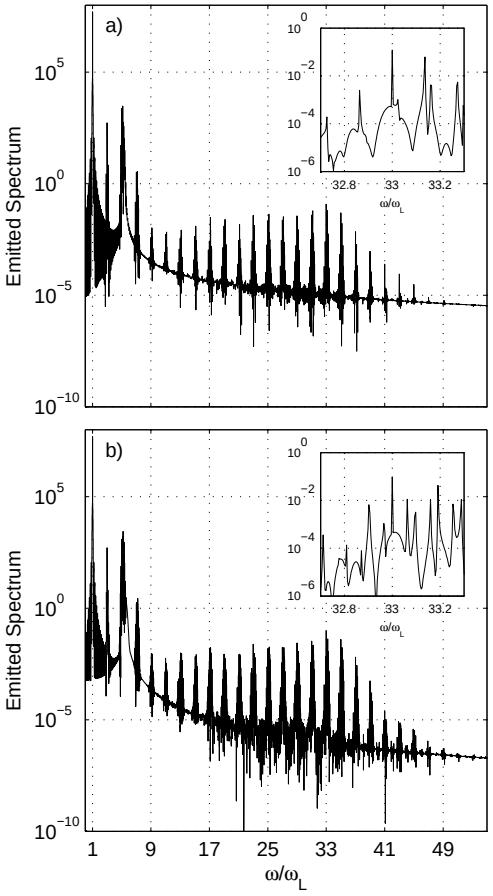


Fig. 1. Emitted spectra by the molecular ions H_2^+ in a) and D_2^+ in b) by using the *semi-classical* model. In the insets there are evident satellite peaks around the usual odd harmonics. The laser intensity is $I_0 = 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm. ω_L is the laser angular frequency.

strong enough to produce HHG and to excite nuclear oscillations around the equilibrium position R_{eq}. For the screening parameters, we have chosen $A_{Ne} = 1.0a_0^2$ and $A_{NN} = 0.03a_0^2$ obtaining $R_{eq} = 2.59a_0$.

In Fig. 1 we show the emitted spectra from the molecular ions by using the *semi-classical* model. In the insets, where we plot part of the spectrum near a particular harmonic, it is evident the presence of satellite peaks surrounding the usual odd harmonics. In particular, the spacing of such sidebands is dependent

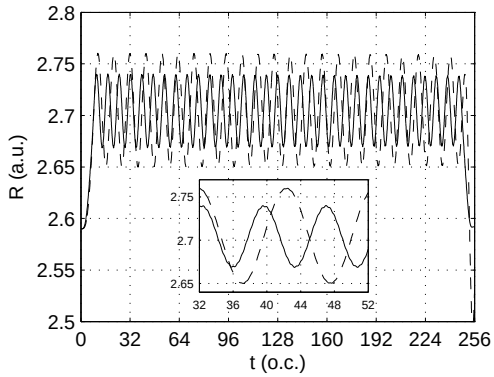


Fig. 2. Temporal evolution of the internuclear distance R of the molecular ions H_2^+ (continues line) and D_2^+(dashed line) by using the *semi-classical* model. The inset shows a particular of the nuclear dynamic. The laser intensity is $I_0 = 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm. R is expressed in term of atomic unit (a.u.)

on the molecular ion considered: the spacing is greater for lighter molecular ion (the H_2^+ nuclear mass is $M_H = 1836.15m_e$; the D_2^+ nuclear mass is $M_D = 2M_H$; m_e is the electron mass). The presence of sidebands has been previously demonstrated in the spectra emitted by molecular ions [10] and attributed to the nuclear motion. In every molecular spectrum the satellite lines are evenly spaced by a particular $\Delta\omega_n$ given by the nuclear vibrational frequency. This is evident by inspecting Fig. 2 where we plot the time evolution of the internuclear distance: the field forces the nuclei to vibrate with regular oscillations at frequency ω_N, different for molecules of different mass. We have numerically checked that these values of ω_N coincides with the spacing $\Delta\omega_n$ observed in the satellite peaks of the emitted radiation. This suggests that it is possible to discriminate among different molecular isotopes by monitoring the radiation emitted by the system.

In order to test if the sidebands depend on the physical model considered, we have investigated the spectra emitted by the molecular ions by using the *full-quantum* model. The results, showed in Figs. 3 and 4, confirm the same characteristics observed with the *semi-classical* model: the presence of satellite peaks around

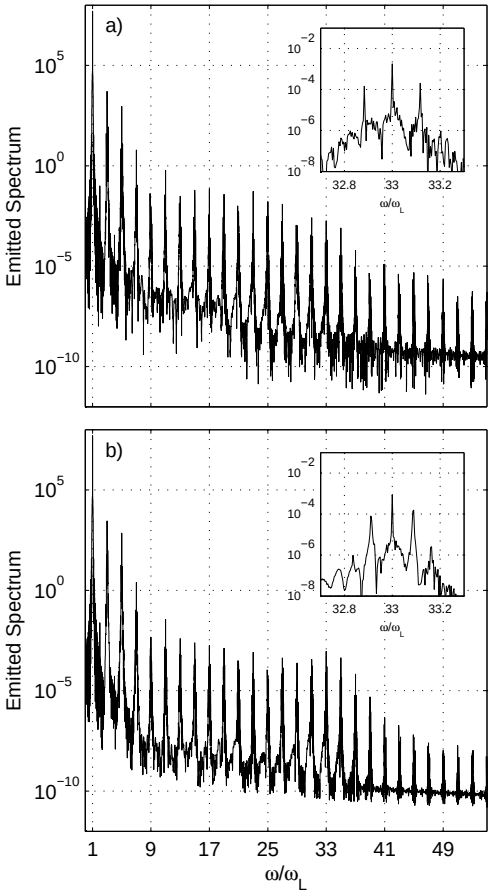


Fig. 3. Emitted spectra by the molecular ions H_2^+ in a) and D_2^+ in b) by using the *full-quantum* model. In the insets there are evident satellite peaks around the usual odd harmonics. The laser intensity is $I_0 = 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm.

the odd harmonics whose frequency spacing corresponds to the oscillation frequency of the quantum averaged internuclear distance R; also, as in the previous model, the spacing depends on the isotope considered.

Furthermore, in order to check the validity of our model and results in a wider context, we have investigated the diffused radiation by two very simple molecules, H_2 and D_2, in similar physical context as above. In this case we have studied the emitted spectra of the molecules by means of the *semi-classical* model and, to

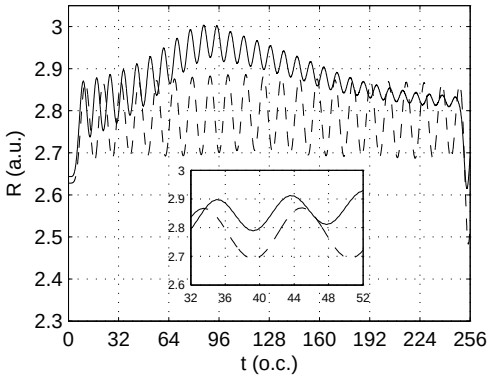


Fig. 4. Temporal evolution of the internuclear distance R of the molecular ions H_2^+ (continues line) and D_2^+(dashed line) by using the *full-quantum* model. The inset shows a particular of the nuclear dynamic. The laser intensity is $I_0 = 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm. R is expressed in term of atomic unit (a.u.)

avoid molecular dissociation, the laser intensity has been reduced to $I_0 = 0.5 \times 10^{14}$ Wcm^{-2}, lower than that used for the molecular ion cases. This is due to the fact that the binding energy of the molecule is less than that of the corresponding ion. The others physical parameters (pulse shape, screening values, spatial box, etc...) unchanged.

In Figs. 5 and 6 we show the emitted spectra and nuclear dynamics for the H_2 and D_2 molecules by using the *semi-classical* model. Again, the spectra are enriched by the presence of the sidebands. Moreover, by comparing the spacing of the satellite peaks in these spectra to the corresponding oscillation frequencies of the nuclei, it is evident a correspondence between them. This confirms that the presence of the satellite peaks is not a typical effect of only molecular ions but also of molecules.

IV. Conclusion

Our numerical simulations show that the motion of the nuclei of simple molecules in intense laser fields introduces in the HHG spectra satellite peaks around the usual odd harmonics. Since such sidebands are absent in the fixed nuclei spectra, they provide a direct evidence of the nuclear vibrational motion. Furthermore,

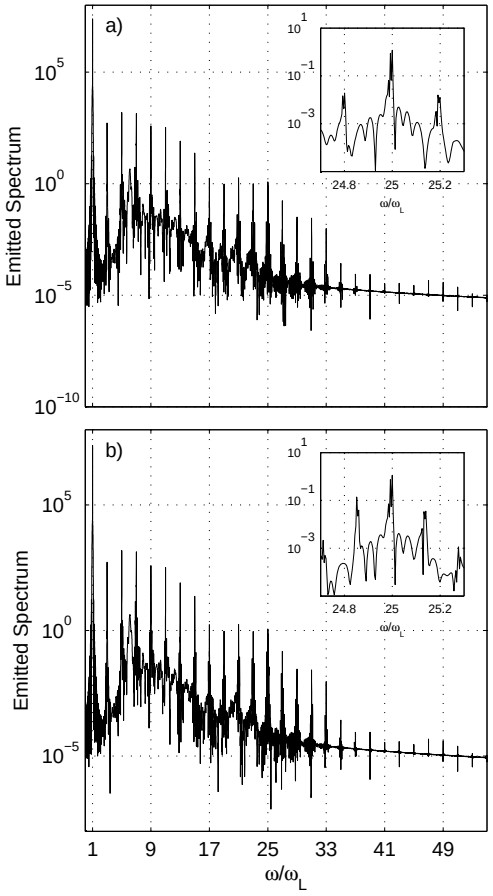


Fig. 5. Emitted spectra by the H_2 in a) and D_2 in b) molecules by using the *semi-classical* model. In the insets there are evident satellite peaks around the usual odd harmonics. The laser intensity is $I_0 = 0.5 \times 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm.

their spacing is given by the oscillation frequency of the nuclei thus allowing a direct measurement of the nuclear oscillation frequency simply by inspection of the HHG spectra. These results are not affected by the quantum nature of the nuclei neither by the presence of more than one electron.

ACKNOWLEDGMENT

This work was partially financed by the University of Palermo Research Program and makes use of results produced by the Project "Electrodynamics Processes in Intense Fields" managed by the "Iniziativa Trasversale Calcolo Parallelo" of CNISM and of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

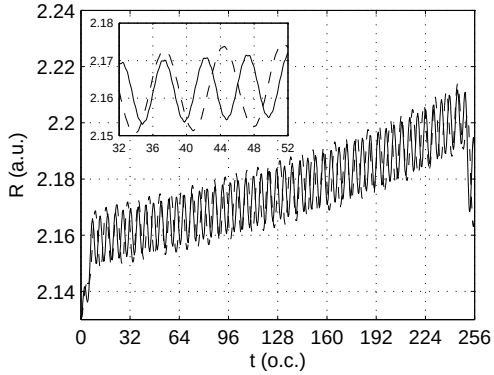


Fig. 6. Temporal evolution of the internuclear distance R of the H_2 (continues line) and D_2(dashed line) molecules by using the *semi-classical* model. The inset shows a particular of the nuclear dynamic. The laser intensity is $I_0 = 0.5 \times 10^{14}$ Wcm^{-2} with a wavelength $\lambda = 780$ nm. R is expressed in term of atomic unit (a.u.)

REFERENCES

[1] B.W. Shore and P.L. Knight, J. Phys. B: At. Mol. Phys. **20**, 413 (1987).
[2] M. Ferray, A. L'Huillier, X. F. Li, L. A. Lompre, G. Mainfray and C. Manus, J. Phys. B: At. Mol. Opt. Phys. **21**, L31 (1988).
[3] X. F. Li, A. L'Huillier, M. Ferray, L. A. Lompre, and G. Mainfray, Phys. Rev. A **39**, 5751 (1989).
[4] P. B. Corkum, Phys. Rev. Lett. **71**, 1994 (1993).
[5] J. Itatani, J. Levesque, D. Zeidler, H. Niikura, H. Pépin, J. C. Kieffer, P. B. Corkum, and D. M. Villeneuve, Nature **432**, 867 (2004).
[6] M. Lein, J. Phys. B: At. Mol. Opt. Phys. **40**, R135 (2007). **39**, S437 (2006). **74**, 023403 (2006).
[7] B. Rotenberg, R. Taïeb, V. Véniard, and A. Maquet, J. Phys. B: At. Mol. Opt. Phys. **35**, L397 (2002).
[8] C. Ruiz, L. Plaja, R. Taïeb, V. Véniard, and A. Maquet, Phys. Rev. A **73**, 063411 (2006).
[9] I. Sundvor, J. P. Hansen, and R. Taïeb, J. Phys. B: At. Mol. Opt. Phys. **39**, 3299 (2007).
[10] P. P. Corso, E. Fiordilino and F. Persico, J. Phys. B. **40**, 1383 (2007).

Three-Dimensional Numerical Model of an H_2^+ Ion in Intense Laser Fields

G. Castiglia[1,3], P.P. Corso[1,3], R. Daniele[1,3], E. Fiordilino[1,3], F. Morales[2,3] and G. Orlando[1,3]

[1]*Dipartimento di Scienze Fisiche ed Astronomiche, Via Archirafi 36, 90123 Palermo, Italy*

[2]*Dipartimento di Fisica e Tecnologie Relative, Viale delle Scienze, 90128 Palermo, Italy*

[3]*Consorzio Nazionale Interuniversitario per le Scienze Fisiche della Materia, via della Vasca Navale 84, 00146 Roma, Italy*

***Abstract*—In this paper we investigate the high order harmonic yields as a function of the alignment angle θ between the axis of the H_2^+ molecular ion and the polarization direction of a driving laser field within the fixed nuclei approximation. We find that the yield of each harmonic has a specific profile as function of the alignment angle and that such a profile is strongly affected by the laser wavelength. Peculiar results are obtained in connection with the attosecond pulse generation. In fact, the attosecond pulses associated to a frequency portion of the emitted spectrum close to the cut-off region (70-80ω_L), present very well defined peaks whose positions are independent of the laser-molecule angle θ. The intensity of these peaks is on the contrary strongly dependent on θ.**

I. Introduction

Atomic or molecular systems, when irradiated by intense laser fields, emit typical spectra consisting of many harmonics of the laser frequency [1], [2], [3], [4]; the envelope of these spectra shows a rapid decrease of the first emitted lines followed by a large plateau of roughly constant amplitude and finally by a rather abrupt cut-off. This phenomenon is usually dubbed high order harmonic generation (HHG). The HHG spectra are usually described in terms of the so called three-step model [5]. During the first step the laser field extracts the active electron out of a bound atomic or molecular state to the continuum. The temporarily free electron drifts along the polarization direction of the laser electric field (second step). When the laser electric field changes its sign, the electron is accelerated back towards the atomic or molecular parent ion; during this re-collision the electron has a certain chance to recombine into a bound state, thus generating the emission of radiation at odd multiples of the laser driving frequency (third step).

Although the harmonics in the plateau region are not phase locked, it has been shown that the single atom emission consists of a train of ultrashort pulses. In fact the signal $S(t)$:

$$S(t) = \left| \int_{\omega_{Min}}^{\omega_{Max}} \boldsymbol{a}_F(\omega) e^{i\omega t} d\omega \right|^2 \qquad (1)$$

where ω_{Min} and ω_{Max} are frequencies in the plateau region, obtained properly filtering the Fourier transform $a_F(\omega)$ of the acceleration $a(t)$ of the electron, consists of two dominant peaks per half laser cycle that are approximately as short as 1/10 of a laser cycle. The generation of attosecond pulses is technologically very important since attosecond pulses may be used for tracing motion of bound electrons or controlling molecular motion [6].

In this paper we analyze in detail the dependence of the HHG spectra emitted by a homonuclear molecular ion on the angle θ between the fixed molecular axis and the laser field. Moreover we show that this θ dependence of

the HHG spectra bears interesting consequences on the generation of high frequency pulses from molecules. In particular, we study the dynamics of the diatomic molecule H_2^+, driven by a laser pulse of frequency ω_L and peak electric field amplitude E_L within the fixed nuclei approximation, i.e. keeping the nuclei fixed [7], [8]. The electron is considered as quantum particle interacting with the fixed nuclei, whose dynamics is described by the Schroedinger equation:

$$i\frac{\partial}{\partial t}\Psi(\boldsymbol{r},t) = H_e(t)\ \ \Psi(\boldsymbol{r},t) \qquad (2)$$

where H_e is the Hamiltonian defined as:

$$H_e = -\frac{\hbar^2}{2m}\nabla_r^2 + V(\boldsymbol{r}) + V_L(\boldsymbol{r},t). \qquad (3)$$

The electron-nuclei interaction energy which appears in equation (3) is

$$V(\boldsymbol{r}) = -\frac{e^2}{[-\boldsymbol{R}/2-\boldsymbol{r}]} - \frac{e^2}{[\boldsymbol{R}/2-\boldsymbol{r}]}. \qquad (4)$$

and the electron-laser interaction

$$V_L(\boldsymbol{r},t) = eF(t)\,\boldsymbol{E}_L\cdot\boldsymbol{r}; \qquad (5)$$

being $\boldsymbol{E}_L = \boldsymbol{E}_0\sin(\omega_L t)$. In all the above equations (3)-(5) $\boldsymbol{r}$ stands for the electron coordinate while $\boldsymbol{R}$ stands for the internuclear distance. In equation (5), which describes the electron-laser interaction potential energy, $\boldsymbol{E}_L$ is the electric field and $F(t)$ is the laser pulse profile.

In the dipole approximation, the emitted radiation of such a system is given by the Larmor formula:

$$P(\omega) = \frac{4e^2}{3c^3}\left|\mathbf{a}_F(\omega)\right|^2 \qquad (6)$$

$\mathbf{a}(\omega)$ being the Fourier transform of the electron acceleration $\mathbf{a}(t)$, obtained by means of the Ehrenfest theorem.

II. Results

The time evolution of the system is obtained by numerically solving the fixed nuclei Schroedinger equation on a spatial grid $512\ a_0$ wide in the x and y electron coordinates and $256\ a_0$ in z, a_0 being the Bohr radius. The values of the grid spacing is $\Delta x = \Delta y = \Delta z = 0.25\ a_0$ and the internuclear distance is $R = 2\ a_0$.

The system is assumed to be in the presence of a laser field with a peak intensity $I_L = 5\times 10^{14}$ W/cm^2. The laser wavelength is λ_L =780 nm, which corresponds to $\omega_L = 0.058$ atomic units (a.u.). With such a choice of the physical parameters the electron ionization signal is small enough to inhibit molecular dissociation so that the approximation of fixed nuclei seems to be justified. This result is shown in Fig.1 where we plot the ionization signal I, defined

$$I = 1 - \int |\psi(\boldsymbol{r},T_{tot})|^2 d^3\boldsymbol{r}, \qquad (7)$$

as a function of the angle θ between the molecular axis and the laser field.

This Figure shows that the ionization signal is very low for any laser orientation for the intensity we are using. Higher ionization signals are obtained when the laser is aligned to the molecular axis; the ionization signal smoothly decreases to a minimum when the laser is perpendicular to such axis.

In Fig. 2 we plot the total emitted spectra for 4 different alignment angles, namely a) 0, b) 30, c) 60 and d) 90 degrees. This Figure shows that the position of the cut-off is roughly independent from the alignment angle and is given by $\omega_{Cut-off} = 80\omega_L$. In Fig.3 we plot the angular dependence of 6 odd harmonics in the plateau region ($\omega = 21\omega_L, 31\omega_L, 41\omega_L, 51\omega_L, 61\omega_L, 71\omega_L$). These plots show the strong dependence of the emitted low peaks from the alignment angle. Furthermore, the angular profile varies from one harmonic to the other. A different situation is found for the harmonic peaks close to the cut-off region, namely those for $71 \le \omega \le 81$, as shown in Fig.4. All the plots in Fig. 4 show a uniform profile for the angular dependence of the emitted peaks: a structure with two lobes, a major lobe when the laser is aligned to the molecular axis and a minor lobe when the laser is perpendicular to this axis. The region for

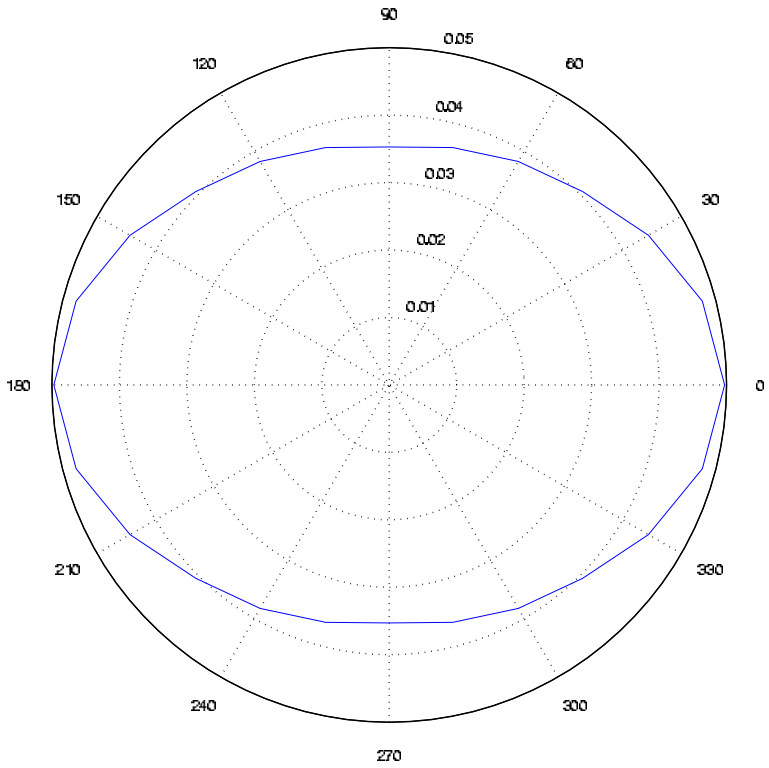


Fig. 1. Angular dependence of the ionization signal.

$30^\circ \leq \theta \leq 75^\circ$ appears as a dark region in terms of emitted harmonics. This uniform dependence of the harmonic peaks in the cutoff region has peculiar consequences in connection with the attosecond pulse generation. Once we fix a frequency interval, $\omega_{Min} \leq \omega \leq \omega_{Max}$, we can evaluate the time profile of the emitted signal associated to such a portion of the spectrum by means of equation (1). In particular, in Fig. 5 we plot the attosecond pulses associated to a frequency portion of the emitted spectrum embedded in the plateau region: $50\omega_L \leq \omega \leq 60\omega_L$. This plot shows a very un-regular profile, for all the alignment angles, with attosecond pulses not very well resolved. On the contrary, the attosecond pulses associated to a frequency portion of the emitted spectrum close to the cut-off region, $70\omega_L \leq \omega \leq 80\omega_L$, represented in Fig. 6, clearly show the presence of very well defined attosecond peaks, regularly spaced. Therefore the form of the emitted attosecond pulse is independent of the angle θ while the intensity of this signal is strongly dependent on the alignment between the field and the molecule.

It is evident that a strong correlation between the temporal evolution of the intensity of the attosecond pulse and the orientation of the molecule. This suggests that the direct observation of the intensity modulation of the emitted attosecond radiation can be used to obtain real-time information about molecular rotation in real experiments where the molecular axis is not fixed. In fact the motion of the nuclei is usually much smaller than that of the radiating electron, therefore the intensity of the attosecond pulse emitted in a given temporal interval, say an optical cycle, can be measured to obtain partial information on the rotational state of the molecular ion.

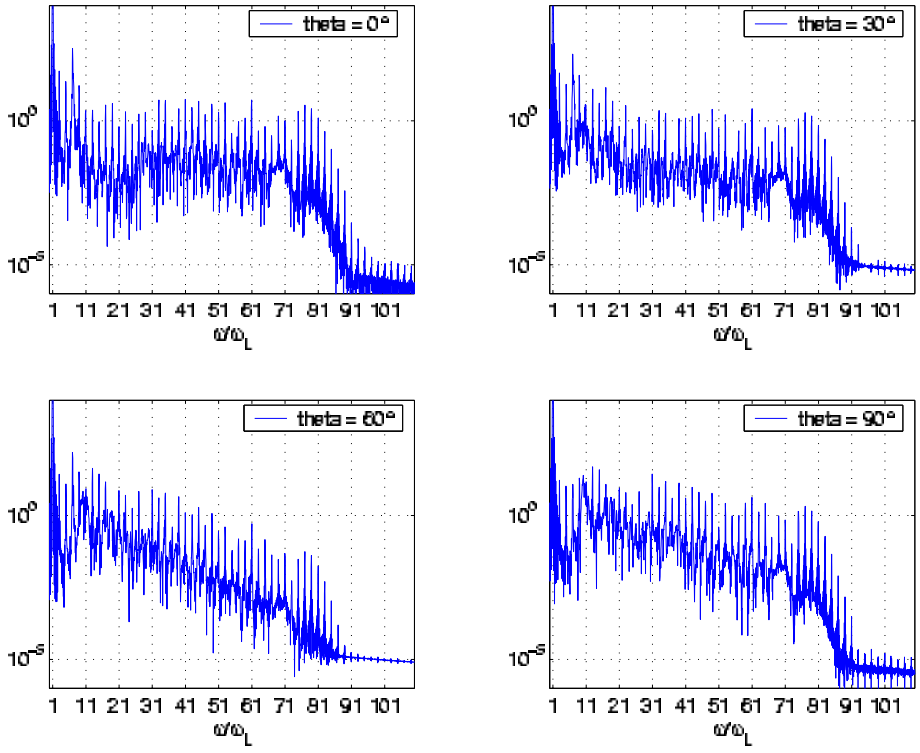


Fig. 2. Total emitted spectra for 4 different alignment angles.

ACKNOWLEDGMENT

This work was partially financed by the University of Palermo Research Program. Moreover this work makes use of results produced by the Project "Electrodynamics Processes in Intense Fields" managed by the "Iniziativa Trasversale Calcolo Parallelo" of CNISM and of the results produced by the PI2S2 Project managed by the Consorzio COMETA, a project cofunded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" (PON 2000-2006) (http://www.pi2s2.it and http://www.consorziocometa.it).

REFERENCES

[1] B.W. Shore and P.L. Knight, J. Phys. B: At. Mol. Phys. **20**, 413 (1987).

[2] Y.I. Salamin, S.X. Hu, K.Z. Hatsargortsyan and C.H. Keitel, Phys. Rep. **427**, 41 (2006).

[3] D.B. Milosevic and F. Ehlotzky, Adv. At. Mol. Opt. Phys. **49**, 373 (2003).

[4] W. Becker et al., Adv. At. Mol. Opt. Phys. **48**, 35 (2002).

[5] P.B. Corkum, Phys. Rev. Letts. **71**, 1994 (1993).

[6] A. Scrinzi , M. Yu Ivanov , R. Kienberger and D.M. Villeneuve J. Phys. B: At. Mol. Opt. Phys. **39** R1 (2006)

[7] Chelkowski S, Zuo T and Bandrauk A D 1992 Phys. Rev. A 46 R5342

[8] Zuo T, Yu H and Bandrauk A D 1996 Phys. Rev. A 54 3290

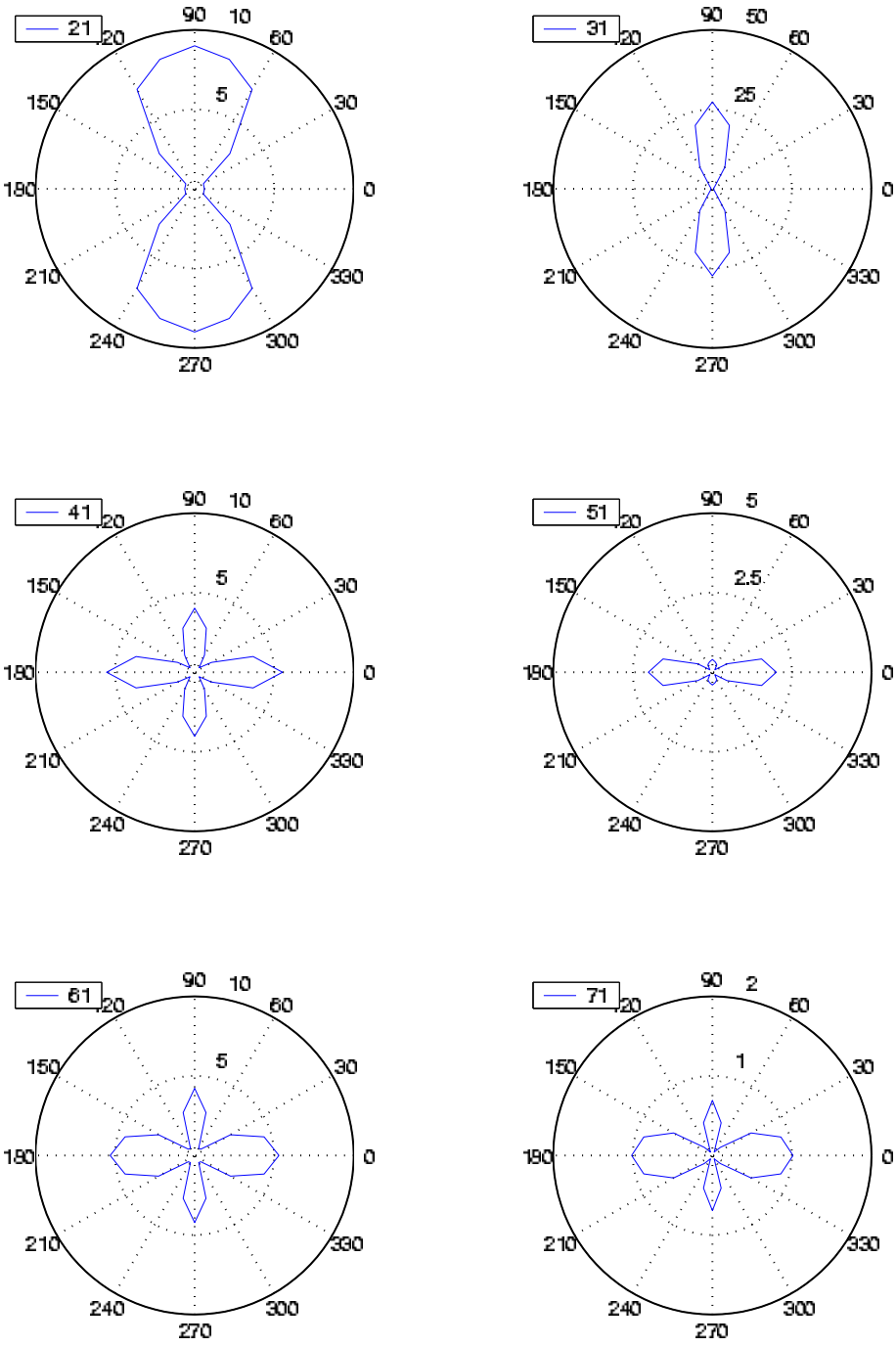


Fig. 3. Angular dependence of six high-order harmonics in the plateau region.

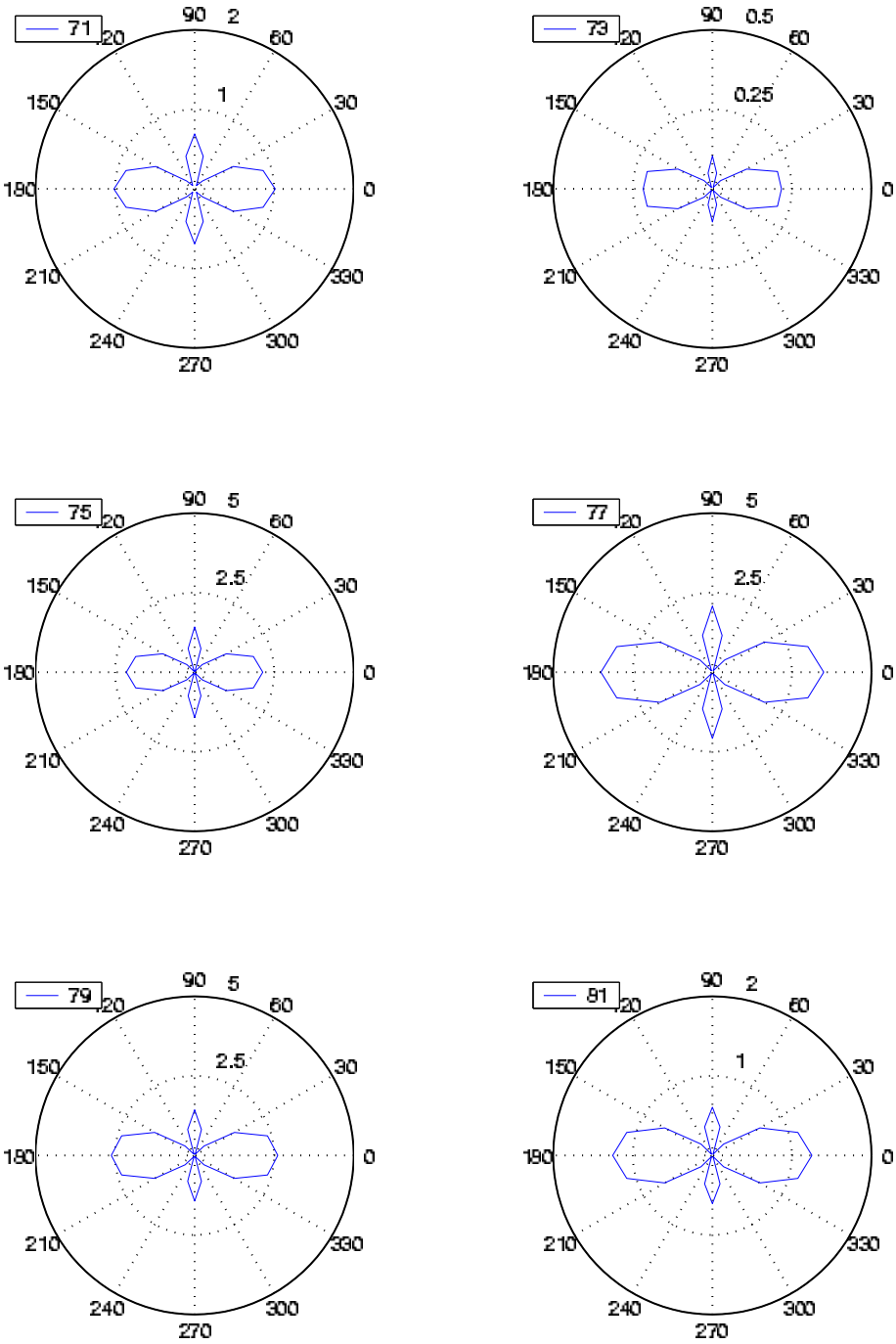


Fig. 4. Angular dependence of six high-order harmonics, basically in the cut-off region.

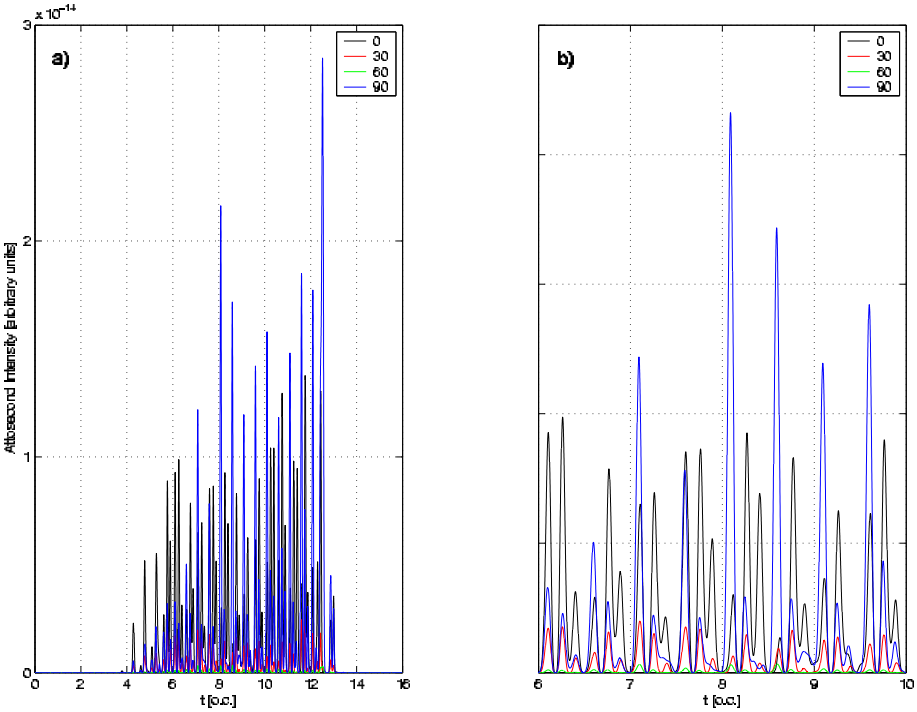


Fig. 5. Attosecond pulses, for different angles, associated to a frequency portion of the emitted spectrum embedded in the plateau region: $50\omega_L \leq \omega \leq 60\omega_L$.

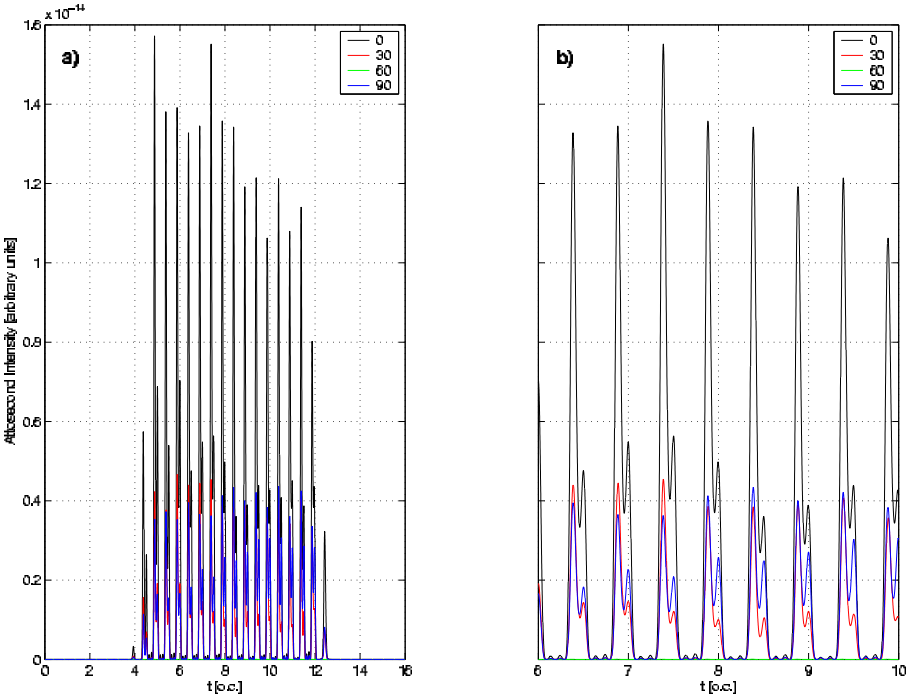


Fig. 6. Attosecond pulses, for different angles, associated to a frequency portion of the emitted spectrum close to the cut-off region: $70\omega_L \leq \omega \leq 80\omega_L$.

Data- and Compute-intensive Applications in Astrophysics: the Virtual Observatory

Fabio Pasian[1,2]

[1]*INAF - Information Systems Unit, Trieste, Italy*
[2]*INAF - Osservatorio Astronomico di Trieste, Trieste, Italy*
pasian@oats.inaf.it

***Abstract* — Astronomy has changed and grown considerably in the past decade, and the new instrumentation is going to further enhance this growth. As a consequence, data archives have become an absolute must to cope with the data avalanche. The effort to integrate and make the various data archives interoperable has been defined as the Virtual Observatory. The goal is to allow the users to feel all astronomical databases to be just "one click away". Achieving this goal would bring a degree of "democratization" in Astronomy, since first-class data are being made available to scientists in developing countries, to students of all levels, to amateurs.**

To make sense, the Virtual Observatory needs to be an international effort, and for this purpose an initiative called the Virtual Observatory Alliance (IVOA) has been built, through the co-ordination of 16 separate national and super-national projects. Involvement at the project but also at the data centre level is required. Interoperability of data and computational resources does not come for free, and work is required at the scientific and technical level to integrate one's own data archive and computing facilities in the VObs and make it interoperable with the others.

The Virtual Observatory concept can be re-used in different domains: initiatives are on-going in this direction and cross-fertilization with other communities has started. Anyway, one must remember that, although the technical details of the VObs development are interesting and challenging, the final goal is Science, i.e. giving astronomers the capability of using new tools and techniques to improve their knowledge and understanding of astro-physical phenomena.

***Index Terms*— Astronomy, Astrophysics, Data Archives, Data Processing, Grid, HPC, Numerical Simulations.**

I. INTRODUCTION

ASTRONOMY in recent years has expanded drastically its observing and modeling capabilities, increasing proportionally its demands for computational power and data access efficiency.

To understand completely the implications of this statement, one should remember that Astronomy is mostly an observational science, where any "experiment" is a snapshot of the fraction of the Universe being observed, and cannot be repeated as such. Astronomy measures $I(\lambda)$, light intensity as a function of wavelength (or frequency, or energy). But, since most phenomena are in fact variable, the measured intensity is $I(\lambda,t)$ which is convolved with the transfer function of the complex system (telescope, instrument, sky) actually involved in the observation. It is therefore easy to understand why every single observation needs to be kept, obviously creating a preservation issue.

The increasing complexity of instrumentation and the need to optimize observing procedures (e.g. the meteo conditions may be unacceptable for one type of observation while suitable for others) lead to a change in concept on modern observatories. Classical observing (the scientist located at the telescope is given the whole

night to perform his/her observations) is replaced by flexible scheduling and service observing, i.e. the observations and the basic data handling are performed by observatory staff. In this case, the scientist owning the observations extracts the data from disk storage, written in the FITS standard data format, which has been used by the community since 1977. Data might have been already partially processed (i.e. calibrated, with removal of the instrument signature).

II. Archives

The scenario described above has a pretty obvious extension, from data storages to full-fledged archives. Observatory policies have been defined so as to allow the data to become public typically after one year from the date of observation. Data taken for a specific scientific purpose can therefore be re-used for different purposes.

This approach, progressively emerged in the past fifteen-twenty years, has allowed the possibility of developing new science. Examples are monitoring the time variability of phenomena, comparing phenomena in different energy bands (the so-called multi-wavelength astronomy), performing statistical analysis or data mining on large quantities of data. From the technical point of view, this implies the possibility of re-processing the raw data when a better know-ledge of instrumental effects is achieved.

An important effect of archives and of the consequent re-use of data, is increasing the return for the investment: since a second of observation on modern facilities is worth roughly 1 US$, and the over-subscription factor is on the average a factor of 5, the possibility of re-using top-class data to do science, or for educational and outreach purposes has also important effects at the level of the society.

But archiving is an activity that is essential to cope with the data avalanche that is hitting the astrophysics community. As a matter of fact, the cumulative photon collecting area of telescopes world-wide has increased by a factor of 8 in the past 25 years, and the detector technology has allowed the production of panoramic chips which have multiplied the data flow. To give an example, the archive of the European Southern Observatory (ESO), the leading European institution, has grown in size a factor of 10 (from 40 to 400 TB) in three years, from 2005 to 2008. And of course, there are world-wide many dozens of telescopes ground- and space-based, covering the full electromagnetic spectrum, and equipped with instruments providing data which are telescope and band-dependent.

III. Computing

Of course, data acquisition and storing is only part of the demand of the astrophysical community for ICT infrastructure: computing holds a great importance.

The first reason is related to the data observed: different levels of processing (reduction, calibration, analysis) are performed. And this requires a lot of resources, especially if data are collected by full-sky surveys.

Not always the data are stored in the archives in "science-ready" form, i.e. with the instrument signature removed. This choice is often made to avoid performing complete re-processing of the data when more precise calibration information is available. In this case, archives may offer as a service the on-the-fly calibration of data at the time they are extracted by the users.

Full-sky surveys produce great quantities of data acquired homogeneously. It is therefore a natural consequence to perform data mining on these data, or statistical analyses.

The needs for computing mentioned above are connected with different types of processing on observational data. As a matter of fact, as in many other disciplines, it is in the field of theory that astrophysics has its most demanding computational tasks. Monte Carlo runs, MFD codes, numerical simulations of phenomena produce requirements for computational capability which may be fulfilled by a variety of computing

architectures: from computer farms, to clusters, from the Grid to full-fledged supercomputers.
An activity where computing merges with intensive data management is the production of data on demand for users. In this case an archive service is integrated by the capability of creating data at the user's wish.
Italy has a good experience and is leading the European astronomical community in porting application software on the Grid [1]. In this conference, some specific examples of data- and compute-intensive applications running on the infrastructures built within the 2000-06 PON projects have been given [2-3]. In the following, two examples of numerical simulations will be briefly summarized.

A. Simulations of the Planck mission

Simulations are an essential step in a project for different reasons: they allow to understand how a given instrument behavior reflects onto the observed sky signal and furthermore are a test bed for the construction of the reduction analysis pipeline of the real data. This yields also information about the size of the problem and gives indication of a proper dimensioning of the whole software and hardware system.

Planck [4] is an all-sky surveyor about to be launched (16 April 2009), and will map the microwave sky in eight frequencies, with the purpose of measuring accurately the anisotropies of the Cosmic Microwave Background (CMB) and of providing some important answers in the field of cosmology.

In order to better understand the instrument behavior, to evaluate the impact of instrumental systematic effects on the scientific results of the mission, and to build and refine the data processing pipeline, a full end-to-end simulation chain as the one depicted in Figure 1 has been devised. The relevant codes are structured as a pipeline composed of several elementary stages, each stage being an algorithm applied onto an input dataset and producing an output dataset. Stages are chained (e.g.. the output of stage A is used as input for stage B).

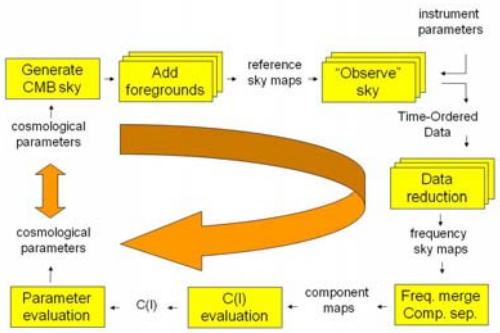


Figure 1 – End-to-end simulations of the Planck mission (explanations in the text).

The following steps are performed: first, the cosmological parameters are used to create the CMB power spectrum, then an CMB map is created from the CMB power spectrum using the HEALPix pixelization scheme [5]. To this map, foreground radiation (synchrotron emission, dust, sources, etc) is added to create "ideal" reference sky maps. These skies are "observed" taking into account mission and instrument parameters (including, e.g,. scanning strategy, beam smearing, detector noise and pointing accuracy) to produce sets of a Time Ordered Data (TOD) consisting of the time sequence of detector outputs, one for every detector. The TOD mimic the observations of the mission. Since knowledge on the mission and instruments increase over time, the whole computational chain must be iterated many times, even after launch.

The "observed" data are then processed to produce maps in each of the Planck frequency bands. The maps are then merged by frequency and analyzed to separate the various components of sky radiation. Once the CMB map has been obtained, the CMB power spectrum is evaluated and its coefficients *C(l)* computed. From the CMB power spectrum, the cosmological parameters can be estimated, and compared with the ones given as an input for the whole procedure, proving a figure of merit

for the accuracy of the data processing chain.

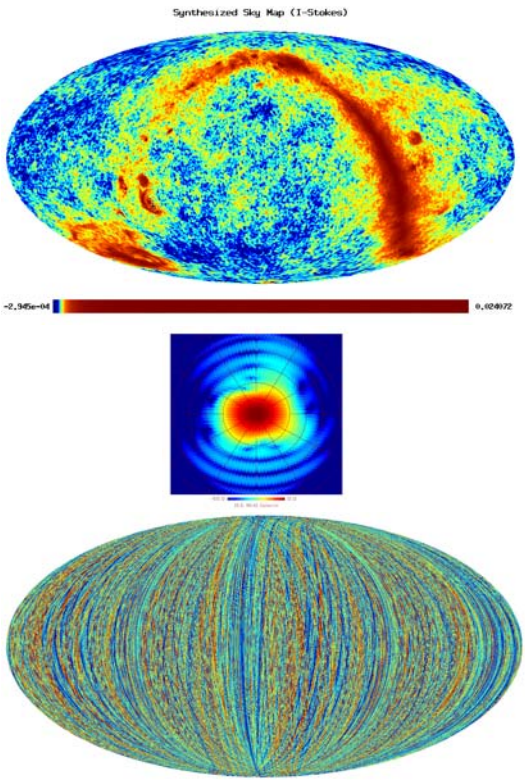


Figure 2 – Evaluation of the effect of beam smearing on observations. From top to bottom an ideal sky map at 30 GHz, a realistic beam for a 30 GHz channel, the residual map obtained subtracting the "observed" map from the ideal one.

The simulations pipeline may be performed only partially to provide specific answers, as shown in Figure 2.

The code is composed of a coarse-grained, embarrassingly parallel part: each detector can run its own "observations" separately, allowing linear scalability with the number of nodes. There is furthermore a finer-grained parallelism which is embedded in the map-making algorithms. The code can be therefore reach high efficiency when performing computations for each detector on a separate high-efficiency cluster.

The Planck simulations code [6] has been already ported on the 400-CPU Planck-LFI DPC local cluster, on the GILDA, Grid.it, EGEE, TriGrid and PI2S2/COMETA Grid infrastructures, on the CINECA, CSC, Jülich and NERSC HPC centers, and on the DEISA HPC Grid [7]. Porting to CRESCO is planned in the next couple of months.

B. Models on demand: the case of BaSTI

BaSTI [8] (a Bag of Stellar Tracks and Isochrones) is a tool designed to provide numerical models for evolutionary tracks, isochrones, luminosity functions, synthetic color-magnitude diagrams, tables with relevant data. BaSTI is also a database/ centre, which provides numerical models on request to astro users. Model production can be heavy from a computational point of view. The database is populated with models produced by the FRANEC code [9].

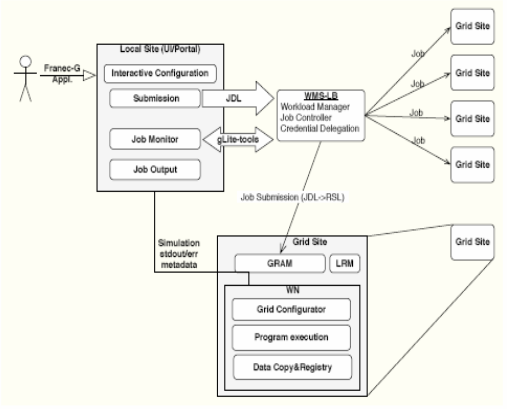


Figure 3 – Mechanism used to port the FRANEC code (FRANEC-G).

FRANEC extensively used for computing models of Stars for a wide range of mass and chemical composition, and in all their evolutionary stages. The user can decide the physical parameters to use (e.g. the equation of state of the star, the opacity, the metallicity, etc.) and select one or more stellar masses to

simulate.

To allow new models to be produced (e.g. to increase the dimensionality of the physical parameters space), FRANEC has been ported to the Grid [10]. The application (FRANEC-G) is designed to hide the complexity of the Grid job submission procedure.

The user interface is now being updated to allow the BaSTI user to query the database for a model having specific physical parameters. If the model is available, it is fed back to the user; if not, the user can choose if he/she wants the new model to be interpolated between existing ones, or to be computed from scratch. In the latter case, the computation is performed on the Grid using FRANEC-G.

IV. Putting it all together: the Virtual Observatory

A. Data and information sources

The sources of data and information for the Astrophysical community are many and can be summarized as follows:

- Telescopes (ground- and space-based, covering the full electromagnetic spectrum) are managed by Observatories
- Instruments (which are telescope and band dependent) are managed by Observatories and/or Consortia
- Data analysis software is usually instrument-dependent: it is managed by Observatories, Consortia and individual Researchers
- Computing is supported by dedicated service Centers, or Organizations that can often be Observatories
- Archives are managed by Observatories and/or Agencies
- Publications are managed by Journals; there is however a single point of search and access, the ADS system
- Data curation, i.e. management of metadata (i.e. the description of data), tables and catalogues) is performed by Data curators
- Public Outreach is managed by Observatories and/or Agencies.

B. The Virtual Observatory

The Virtual Observatory (VObs) is an innovative, still evolving, system to take advantage of astronomical data explosion (e.g., use statistical identification to perform multi-wavelength, multi-parameter analysis, to allow astronomers to interrogate multiple data centers in a seamless and transparent way and to utilize at best astronomical data, to permit remote computing and data analysis and to foster *new science*.

The VObs aims at achieving the goal of having *all astronomical databases inside the scientist's PC* just as in the Web all documents are easily retrievable inside one's own PC.

Figure 4 – Interoperability among the information elements in the astrophysics community through the Virtual Observatory (from [11]).

The situation depicted in the above section is pretty complex, and the desire has been felt to unify the sources of information within a federated framework, as depicted in Figure 4 above.

Achieving the ambitious goal of allowing the different sources of information to interoperate is not trivial: as a matter of fact, the current situation shows both positive and negative aspects.

On the positive side, one can consider that observational data are freely available *on-line* in general 1 year after observation; both data

and catalogues are normally stored in *astronomical archives*. Results are published in academic journals, all *available on-line* with *pointers* to data archives; as mentioned above, there is one *single entry point for journals* (ADS). The processing and analysis software *maintained* Observatories/Archives is made available *on-line*.

However, the different astronomical archives have *widely different access/search interfaces and standards/conventions*; and they mainly serve raw data. The widely specialized, *analysis software* for the various sub-branches is in general *complex*, thus yielding a *steep learning curve*, but this is needed, since multi-wavelength is now the norm to produce science. Publication-Archive links often point to *raw, unprocessed data* and are not complete Furthermore, object *metadata are not homo-geneously defined*.

C. The International Virtual Observatory Alliance

All of these considerations require that the various players speak the same language. It is therefore necessary that common standards and protocols are defined and adopted within the whole community. This issue is tackled within the International Virtual Observatory Alliance (IVOA), which coordinates 16 world-wide national VObs projects.

The IVOA mission is *"to facilitate the international coordination and collaboration necessary for the development and deployment of the tools, systems and organizational structures necessary to enable the international utilization of astronomical archives as an integrated and interoperating virtual observatory".* The work is carried out by means of teleconferences, "TWiki" pages, and bi-annual meetings (the last one was carried out in Baltimore in October 2008, the next will be in Strasbourg in May 2009).

The IVOA goals are the standardization of data, metadata and software, the definition of data interoperability methods, and the production of lists of available data and computing services (each provided by the individual VObs projects).

Figure 5 – IVOA member organizations.

From the organizational point of view, the IVOA structure is composed of an Executive Board which includes the representatives from all national VObs projects**,** a dozen of Working and Interest Groups coordinated by a Technical Coordination Group. There is a total of 400-500 individuals involved in IVOA activities.

IVOA is an initiative that is based on the concept of collaborating and sharing. The absence of specific funding does not allow to force any decisions, which are basically taken unanimously. This practice is in principle good, since it builds consensus in the community. However, competition at the personal and/or project level brings the consequence of a slow convergence in the definition of standards.

D. Extending the Virtual Observatory concept

In the last couple of years, the VObs received support at an International level: positive statements on the importance of the initiative were made by the Organization for Economic Co-operation and Development (OECD), by the European Strategy Forum on Research Infrastructures (ESFRI), and by ASTRONET, the European initiative aiming at

enhancing the coordination and cooperation between national funding and research management organizations in Europe who are responsible for astronomical research. This recognition and support is important since, to make sense, the Virtual Observatory needs to be an international effort.

In this framework, an important comment was made in the 2006 ESFRI document, where the Virtual Observatory was indicated as the example to follow to create an international federation of libraries. As a matter of fact, the VObs concepts can be, and are being, re-used in different domains. Among the various interested communities, it is worth mentioning that the planetologists have their own International Planetary Data Alliance (IPDA), the Solar community has its own VObs and DataGrid, and the High-Energy Physics (HEP) community is interested in the VObs mechanisms, as evidenced in the "HEP Data Preservation Workshop" held in Hamburg in January 2009 [12].

V. Generalization

The documents of the ASTRONET initiative [13,14] have defined the Virtual Observatory as *"the e-infrastructure project in Europe"*. As a matter of fact, an e-infrastructure is a mechanism that, based on solid networking foundations, allows Computing, Data and Applications of various kinds (Data Reduction, Data Analysis, Theory, Numerical Simulations) to interoperate. Interoperability is also the key word of the Virtual Observatory.

Integration of the VObs with computational infrastructures, and in particular with Grid facilities, is of key importance. Effort in this direction has been carried out, also within EU-funded projects. In particular, specific IVOA standards have been developed on top of Grid middleware: a FITS driver has been built on top of *gridftp* [15]; the IVOA Single Sign-On standard has been implemented by means of the Grid Authentication & Authorization tools, VOSpace on top of the Data Management mechanisms, VObs Workflows using Job Management, VObs registries using the Grid Information Systems [16]. Furthermore, a "native" way of accessing databases from the Grid through a Query Element (similar in structure to the CE) was developed [17].

The Virtual Observatory has an even more ambitious goal: by allowing high-level applications to interoperate and exchange information, it is possible to build a Knowledge Infrastructure [18].

In any case, it is important to always remember that an e-infrastructure is a service for the scientists. For this reason, it is important to simplify as much as possible the interface to data and computing services for the scientist.

Figure 6 – The user needs to be guaranteed access to computing and data services transparently and through a unique interface which, for astronomy, is the Virtual Observatory.

The scientist is interested in producing top-class science, and is not particularly fond of knowing the details of the data structures he/she is using, or the mechanisms governing parallel computations, or the nuts and bolts of the applications. Therefore, the goal in building an e-infrastructure for astronomy is to hide in a "cloud" all of these details, and allow the scientist to have a single user interface to all this.

This goal is ambitious and requires lots of standardization efforts at an international level. This is the direction that the Virtual Observatory is following, to support the community towards the production of science.

VI. Conclusion

Astronomy has changed and grown considerably in the past decade, and the new instrumentation is going to further enhance this growth. As a consequence, data archives have become an absolute must to cope with the data avalanche. The effort to integrate and make the various data archives interoperable has been defined as the Virtual Observatory. The goal is to allow the users to feel all astronomical databases to be just "one click away". Achieving this goal would bring a degree of "democratization" in Astronomy, since first-class data are being made available to scientists in developing countries, to students of all levels, to amateurs.

To make sense, the Virtual Observatory needs to be an international effort, which requires involvement at the *project* but also at the *data centre* level. This does not come for free, and work is required at the scientific and technical level to integrate one's own data archive in the VObs and make it interoperable with the others.

The Virtual Observatory concept can be re-used in different domains: initiatives are on-going in this direction and cross-fertilization with other communities have started.

Anyway, one must remember that, although the technical details of the VObs development are interesting and challenging, the final goal is science, i.e. giving astronomers the capability of using new tools and techniques to improve their knowledge and understanding of astro-physical phenomena.

Acknowledgments

The author would like to thank many friends and colleagues who have influenced his views on the Virtual Observatory and the Grid: Paolo Padovani, Françoise Genova, Dave De Young, Christophe Arviset, Claudio Vuerli, Ugo Becciani, Santi Cassisi, Giuliano Taffoni, Giuseppe Longo, Piero Benvenuti, Charles Loomis, Mirco Mazzucato, Roberto Barbera.

References

[1] Benacchio L., Pasian F. (eds.), "Grid-enabled astrophysics", Polimetrica International Scientific Publisher, 2007.

[2] Antonuccio-Delogu V., "Project GALCOS: GALaxy formation within a COSmological volume", these Proceedings.

[3] Brescia M., "The DAME/VO-Neural infrastructure: an integrated data mining system support for the e-science community", these Proceedings.

[4] Tauber J. A., "The Planck Mission", in New Cosmo-logical Data and the Values of the Fundamental Parameters, Proceedings of IAU Symposium #201, Edited by A. Lasenby and A. Wilkinson. San Francisco, CA: Astronomical Society of the Pacific (ASP), 2005., p.86.

[5] Górski, K., M.,Hivon, E.,Banday, A. J., Wandelt, B. D., Hansen, F. K., Reinecke, M.; Bartelmann, M., "HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere", The Astrophysical Journal, Volume 622, Issue 2, 2005, p. 759.

[6] Reinecke, M.; Dolag, K.; Hell, R.; Bartelmann, M.; Enßlin, T. A., "A simulation pipeline for the Planck mission" Astronomy and Astrophysics, Volume 445, Issue 1, 2006, p.373.

[7] Gheller, C., Leach, S., Taffoni, G., Vuerli, C., Pasian, F., "Simulating the Planck Mission on a Distributed Computing Architecture: The DEISA and EGEE Experiences", Astronomical Data Analysis Software and Systems XVI ASP Conference Series, Vol. 376, Edited by Richard A. Shaw, Frank Hill and David J. Bell., 2007, p.61.

[8] Cassisi, S., Pietrinferni, A., Salaris, M., Castelli, F., Cordier, D., Castellani, M., "BASTI: an interactive database of updated stellar evolution models", Memorie della Società Astronomica Italiana, v.77, 2006, p.71

[9] Degl'Innocenti, S.,Prada Moroni, P. G.; Marconi, M., Ruoppo, A., "The FRANEC stellar evolutionary code", Astrophysics and Space Science, Volume 316, Issue 1-4, 2008, p. 25.

[10] Pasian, F.; Taffoni, G.; Vuerli, C., Manzato, P.; Gasparo, F., Cassisi, S., Pietrinferni, A.; Salaris, M., "BaSTI: the Virtual Observatory meets the Computational Grid", Astronomical Data Analysis Software and Systems ASP Conference Series, Vol. 394, Edited by Robert W. Argyle, Peter S. Bunclark, and James R. Lewis, 2008, p.285.

[11] Quinn, P. J.; Barnes, D. G.; Csabai, I.; Cui, C.; Genova, F.; Hanisch, R.; Kembhavi, A.; Kim, S. C.; Lawrence, .; Malkov, O., Ohishi, M.; Pasian, F.; Schade, D.; Voges, W., "The International Virtual Observatory Alliance: recent technical developments and the road ahead", Optimizing Scientific Return for Astronomy through Information Technologies. Edited by Quinn, Peter J.; Bridger, Alan. Proceedings of the SPIE, Volume 5493, 2004, p. 137

[12] First Workshop on Data preservation and Long Term Analysis in HEP, Proceedings edited by Christinel Diaconu, in press.
[13] "A Science Vision for European Astronomy", ASTRONET document, Editors P.T. de Zeeuw & F.J. Molster, ISBN 978-3-923524-62-4, 2007.
[14] "The ASTRONET Infrastructure Roadmap: A Strategic Plan for Europan Astronomy", ASTRONET document, Editors: Michael F. Bode, Maria J. Cruz & Frank J. Molster, ISBN: 978-3-923524-63-1, 2008.
[15] Taffoni, G.; Barisani, A.; Vuerli, C.; Pence, W.; Pasian, F., "Grid Data Source Engine: Grid gateway to the Virtual Observatory", Astronomical Data Analysis Software and Systems XV ASP Conference Series, Vol. 351, Edited by Carlos Gabriel, Christophe Arviset, Daniel Ponz, and Enrique Solano, Astronomical Society of the Pacific, 2006, p.508.
[16] Taffoni, G.; Vuerli, C.; Pasian, F.; Rixon, G., "Bridging VO and Computational Grid Applications within Euro-VO", Astronomical Data Analysis Software and Systems ASP Conference Series, Vol. 394, Edited by Robert W. Argyle, Peter S. Bunclark, and James R. Lewis, 2008, p.289.
[17] Taffoni, G.; Vuerli, C.; Barisani, A.; Smareglia, R.; Volpato, A.; Pastore, S.; Baruffolo, A.; Pasian, F.; Benacchio, L.; Ambrosi, E.; Ghiselli, A., "Grid Data Source Engine: Grid gateway to the Virtual Observatory", Astronomical Data Analysis Software and Systems XV ASP Conference Series, Vol. 351, Edited by Carlos Gabriel, Christophe Arviset, Daniel Ponz, and Enrique Solano. San Francisco: Astronomical Society of the Pacific, 2006., p.508.
[18] Pasian F., "HPC, Grid and Data Infrastructures for Astrophysics: an Integrated View", Il Nuovo Cimento, 2009, in press

Fabio Pasian is Chair of the International Virtual Observatory Alliance (IVOA), the world-wide collaboration aimed at coordinating the implementation of transparent access to data archives and at defining the related standards. Since June 2004 he has been leading the Information Systems Unit of the Italian National Institute of Astrophysics (INAF).
He has led several national and international projects on the following topics: design of astronomical databases and archives, data processing for space missions, Grid, Virtual Observatory. He has done consultant activity for ESA and ESO in the fields of information systems and data processing and archiving.
Currently, he coordinates the Italian participation (VObs.it) in IVOA and is the Italian representative in the IVOA Executive Committee. He is the Italian coordinator of two VObs-related projects funded within the EU Sixth Framework Programme. He is also Co-I for the LFI instrument of the ESA Planck mission, and has responsibilities in the design of its Data Processing Centre.

ISBN 978-88-95892-02-3