
Progetto CRESCO

Centro computazionale di RicErca sui Sistemi Complessi

SP IIL.3 Modelli e Strumenti di supporto alla Ottimizzazione e Riconfigurazione delle Reti

Deliverable 3

Report sugli algoritmi di instradamento del modello di telecomunicazioni

Unità operativa DIIMA

Dipartimento di Ingegneria dell'Informazione e Matematica Applicata
Università di Salerno



27/05/2008

Responsabile Unità Operativa
Prof. Ciro D'Apice

SOMMARIO

Introduzione.....	3
1 Modello fluidodinamico per reti di telecomunicazioni.....	4
2 Principi di instradamento	7
2.1 Algoritmo di instradamento basato sullo stato del link.....	10
2.2 Algoritmo di instradamento distance vector	12
2.3 Algoritmi RA1 e RA2	13
3 Definizione degli algoritmi di instradamento nel modello fluidodinamico.....	17
3.1 Nozioni di base.....	17
3.2 Dinamiche sulle linee.....	18
3.3 Dinamiche sulla rete.....	21
3.4 Distribuzione di traffico alle giunzioni	25
3.5 Riemann solvers alle giunzioni	28
3.5.1 Dinamiche ai nodi secondo l'RA1	29
3.5.2 Dinamiche ai nodi secondo l'RA2	32
4 Bibliografia	38

Introduzione

Le attività previste dal Dipartimento di Ingegneria dell'Informazione e Matematica Applicata dell'Università di Salerno nell'ambito del progetto Cresco, inquadrato nel **SP III.3 Modelli e Strumenti di supporto alla Ottimizzazione e Riconfigurazione delle Reti**, hanno riguardato la definizione di algoritmi di instradamento per reti di telecomunicazioni modellate con un approccio fluidodinamico, basato sulle leggi di conservazione.

Il documento è articolato in tre parti. La prima riporta brevemente il modello macroscopico proposto per reti di telecomunicazioni e gli algoritmi di instradamento utilizzati: RA1 e RA2. In particolare, vengono evidenziate in maniera sostanziale le principali differenze tra i due algoritmi di routing: il primo instrada i pacchetti sui link di uscita in base a dei coefficienti prefissati; il secondo, invece, tiene conto di possibili sovraccarichi della rete prevedendo il reindirizzamento dei pacchetti. Viene trattato anche il fenomeno dei loops, in cui potrebbero incorrere i pacchetti nel caso del secondo algoritmo, RA2, problema risolto mediante l'introduzione, nel modello, delle sorgenti e delle destinazioni.

La seconda parte analizza la stratificazione a livelli della rete Internet, spiegando come il modello sia in grado di descrivere applicazioni sia al livello trasporto sia allo strato di rete, analizza i principi di instradamento generali relativi alle reti di telecomunicazioni, e in particolare gli algoritmi di instradamento che si usano nel modello fluidodinamico. Nello specifico, nella seconda sezione, si riporta una descrizione di alcuni algoritmi di instradamento globale, un cui esempio è l'algoritmo di Dijkstra, e di algoritmi di instradamento decentralizzato, come l'algoritmo distance vector. Il funzionamento di tali algoritmi ricalca le procedure di instradamento usate nel modello fluidodinamico per reti di telecomunicazioni.

La terza ed ultima parte del documento, invece, riporta una descrizione rigorosamente analitica degli algoritmi di instradamento. Nello specifico, viene data dapprima una panoramica sulle grandezze fondamentali del modello, si illustrano le dinamiche dei pacchetti sulle linee della rete di telecomunicazioni, e poi sulla rete e quindi sui nodi, e si introduce un'equazione semilineare che descrive le funzioni tipo – traffico, che misurano la porzione di densità totale proveniente da una sorgente e che va verso una destinazione. A partire poi dalla definizione dei Riemann Solver per la descrizione della dinamica ai nodi della rete, viene presentata una formulazione matematicamente esaustiva degli algoritmi di instradamento proposti.

1 Modello fluidodinamico per reti di telecomunicazioni

La rete di telecomunicazioni viene descritta da un numero finito di linee di trasmissione modellate da intervalli $[a_i, b_i]$ (con uno dei due estremi possibilmente infinito) che si incontrano in qualche nodo (routers) che riceve e invia informazioni in forma di pacchetti. Ciascun pacchetto può essere visto come una particella sulla rete, ma bisogna tener conto di questioni specifiche di telecomunicazioni, ad esempio avendo in mente la rete Internet si assume che ciascun pacchetto viaggia sulla rete ad una fissata velocità e con un'assegnata destinazione finale e che i routers ricevono, processano e instradano i pacchetti che possono essere persi con una probabilità che aumenta all'aumentare dei pacchetti processati.

L'idea è di guardare alla rete in una scala temporale intermedia in modo tale che la trasmissione dei pacchetti avviene ad un livello più veloce ma gli equilibri della intera rete sono raggiunti solo asintoticamente. Ciò permette di costruire un modello che si basa su una descrizione macroscopica. Se si considerano due routers consecutivi, poichè ciascun router rimanda al nodo successivo i pacchetti persi fin quando non arriva ad esso il segnale di ricezione, si può assumere che in una scala temporale intermedia i pacchetti sono conservati. Passando poi a "backbones" della rete, che sono formati da molti router consecutivi, e guardando in una scala spaziale larga, si arriva a considerare un modello consistente in una singola legge di conservazione, in cui la quantità conservata è la densità di pacchetto:

$$\rho_t + f(\rho)_x = 0,$$

dove ρ è la densità dei pacchetti (numero di pacchetti per unità di spazio) ed $f = \rho v(\rho)$ è il flusso (numero di pacchetti per unità di tempo), con $v(\rho)$ velocità di trasmissione dei pacchetti.

Poiché la velocità su ciascuna linea è supposta costante, è possibile ricavare una velocità media di trasmissione tra i routers considerando la quantità di pacchetti che può essere persa. Assegnando una probabilità di perdita di pacchetto come funzione della densità, passando al limite nella procedura di invio e rinvio dei pacchetti si ottiene una funzione velocità e quindi una funzione flusso.

Si è supposta una funzione di probabilità di perdita di pacchetti del tipo:

$$p(\rho) = \begin{cases} 0, & 0 \leq \rho \leq \sigma, \\ \frac{2(\rho - \sigma)}{\rho}, & \sigma \leq \rho \leq \rho_{\max}, \end{cases}$$

dove $\rho_{\max}$ è la densità massima supportata da una linea, e σ è un valore soglia prefissato.

A partire da tale probabilità si è ricavata una velocità media di trasmissione

$$v(\rho) = \bar{v}(1 - p(\rho)) = \begin{cases} \bar{v}, & 0 \leq \rho \leq \sigma, \\ \bar{v} \frac{(2\sigma - \rho)}{\rho}, & \sigma \leq \rho \leq \rho_{\max}, \end{cases}$$

e da qui la funzione flusso.

Su ciascuna linea di trasmissione viene anche considerato un vettore π che descrive le funzioni di tipo traffico, cioè le percentuali di pacchetti che vanno da una fissata origine ad una fissata destinazione e la cui introduzione è giustificata in seguito. L'evoluzione di π segue un'equazione semilineare

$$\pi_t + v(\rho)\pi_x = 0,$$

che viene accoppiata alla legge di conservazione.

Poiché l'obiettivo è quello di modellare reti complesse è stata risolta la dinamica ai nodi in cui molte linee si intersecano. Il passo principale per la risoluzione del Problema di Cauchy ai nodi consiste nella risoluzione del problema di Riemann, problema di Cauchy con dati iniziali di tipo Heavy-side. Alle giunzioni i Problemi di Riemann sono sottodeterminati per cui sono stati introdotti ulteriori condizioni. Sono state considerati due algoritmi di instradamento dei pacchetti:

RA1: I pacchetti dalle linee entranti vanno su quelle uscenti in base alla destinazione finale, massimizzando il numero di pacchetti solo sulle linee entranti.

RA2: I pacchetti dalle linee entranti vanno instradati su quelle uscenti al fine di massimizzare il numero di pacchetti sulle linee entranti ed uscenti.

Uno dei drawbacks dei precedenti algoritmi è che non si considera il percorso globale dei pacchetti. Per esempio consideriamo una rete in cui alcuni nodi sono congestionati: se usiamo l'algoritmo RA2 che massimizza il flusso sulle linee entranti ed uscenti i pacchetti non sono instradati verso nodi congestionati, per cui potrebbero entrare in loops.

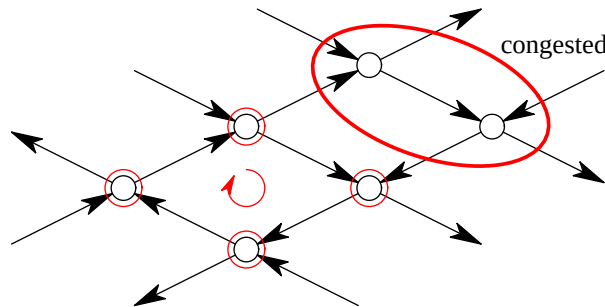


Figura 2.1: Rete congestionata e loops.

Questi ultimi sono evitati se si considerano che i pacchetti originati da una sorgente e con un'assegnata destinazione hanno percorsi precisi all'interno della rete. Tali percorsi sono determinati dal comportamento alle giunzioni attraverso il vettore π .

Sono state considerate diverse funzioni di distribuzioni del traffico che descrivono diverse strategie di routing:

- 1) Ad una giunzione il traffico iniziato da una sorgente s e con destinazione finale d , proveniente dalla linea di trasmissione i , viene instradato su un'assegnata linea j ;
- 2) Ad una giunzione il traffico iniziato da una sorgente s e con destinazione finale d , proveniente dalla linea di trasmissione i , viene instradato su ogni linea (con coefficienti probabilistici) o su alcune di esse.

La prima funzione di distribuzione del traffico era già stata considerata per il traffico stradale, per cui ci è focalizzati sulla seconda, in particolare sono stati analizzate due casi:

- 2.a) in una giunzione il traffico proveniente da un'assegnata sorgente e con un'assegnata destinazione viene instradato o non instradato su una linea uscente;
- 2.b) in una giunzione il traffico proveniente da un'assegnata sorgente e con un'assegnata destinazione viene instradato su ogni linea uscente con una certa probabilità.

Partendo da una funzione di distribuzione del traffico ed usando il vettore π , si calcola la matrice di distribuzione del traffico.

Per risolvere la dinamica ai nodi infine sono state definite le soluzioni ai Problemi di Riemann nei seguenti casi:

- Algoritmo RA1, funzione di distribuzione di tipo 2a);
- Algoritmo RA1, funzione di distribuzione di tipo 2b);
- Algoritmo RA2 funzione di distribuzione di tipo 2a);
- Algoritmo RA2 funzione di distribuzione di tipo 2b).

2 Principi di instradamento

Una rete può essere stratificata a livelli, come illustrato nella seguente figura:

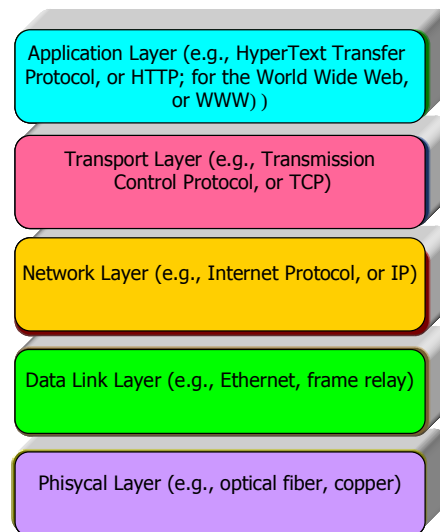


Figura 2.1: stratificazione della rete Internet.

Analizziamo nel dettaglio lo strato di trasporto e lo strato di rete.

Lo strato di trasporto permette la regolazione del flusso di informazioni mediante protocolli affidabili (TCP/IP) che gestiscono meccanismi di acknowledgement (corretta ricezione dell'informazione) e ritrasmissione (in caso di perdita di informazioni).

Lo strato di rete gestisce la comunicazione tra una macchina (host) ed un'altra; accetta una richiesta di inoltro di un pacchetto dallo strato di trasporto; crea il datagram di base della rete formattando opportunamente il pacchetto ricevuto per poi trasferirlo dalla sorgente alla destinazione senza garantire la correttezza della trasmissione.

Quindi, la differenza principale tra i due strati è che:

- lo strato di trasporto è affidabile e garantisce la conservazione del numero dei pacchetti (con opportuni meccanismi di controllo dei pacchetti ricevuti e ritrasmissione);
- lo strato di rete non è affidabile e smista le informazioni ricevute dal livello trasporto dalla sorgente alla destinazione. Di fatto, la non affidabilità dei protocolli dello strato di rete non è sorprendente, dal momento che eventuali controlli per una corretta ritrasmissione si sono già avuti al livello di trasporto.

Il modello fluidodinamico proposto per le reti di telecomunicazioni è basato sulla conservazione del numero dei pacchetti sulle linee della rete. Sorge, quindi, l'esigenza di far riferimento a comunicazioni affidabili. Quindi, il modello può essere usato in tutti quei tipi di comunicazioni che fanno uso del protocollo TCP/IP.

La legge di conservazione che descrive il modello può essere utilizzata per definire le dinamiche su una linea di trasmissione; ma, affinché possa descrivere le dinamiche ai nodi, occorrono delle regole, che sono, appunto, gli algoritmi di instradamento, che regolano il passaggio dell'informazione da un punto all'altro della rete. Quindi, per la descrizione delle dinamiche ai nodi, si può far riferimento allo strato di rete.

Il ruolo dello strato di rete è “ingannevolmente” semplice: trasportare pacchetti da un host che spedisce ad un host che riceve, e quindi regolare dinamicamente i meccanismi che intervengono nel trasporto dell'informazione tra sorgente e destinazione. I ruoli che competono a questo strato sono principalmente tre:

- *determinazione del percorso.* Lo strato di rete deve determinare il tragitto seguito dai pacchetti che fluiscono da un sender a un receiver. Gli algoritmi che calcolano questi percorsi sono detti **algoritmi di instradamento** (*routing algorithms*). Un algoritmo di instradamento deve determinare, per esempio, il percorso lungo il quale i pacchetti devono fluire da una certa sorgente (host sorgente) H_1 ad una destinazione (host destinazione) H_2 ;
- *commutazione.* Quando un pacchetto giunge all'ingresso di un router, questo deve trasferirlo sull'appropriato link (linea) in uscita. Per esempio, un pacchetto in arrivo dall'host H_1 al router R_2 deve essere instradato al successivo router sul percorso verso H_2 .
- *impostazione della chiamata.* Nel protocollo TCP, è necessario un handshake a tre vie (un meccanismo di controllo incrociato) prima che i dati fluiscono realmente da sender a receiver. Questo permette a sender e receiver di impostare le necessarie informazioni di stato (per esempio, numeri di sequenza e valore iniziale della finestra per il controllo del flusso). In modo analogo, alcune architetture dello strato di rete (per esempio, il protocollo ATM) richiedono che i router lungo il percorso scelto dalla sorgente alla destinazione scambino un handshake (un meccanismo di sincronismo) in modo da impostare lo stato prima che i dati iniziano effettivamente a fluire. Nello strato di rete, a questo processo ci si riferisce come all'**impostazione della chiamata** (call setup).

Analizziamo più nel dettaglio gli algoritmi di instradamento.

Per trasferire i pacchetti da un host che spedisce all'host di destinazione, lo strato di rete deve determinare il percorso (path) o rotta (route) che i pacchetti devono seguire. A prescindere dal fatto che lo strato di rete può fornire un servizio datagram (in questo caso differenti pacchetti fra una coppia di host possono seguire percorsi diversi) o un servizio a circuito virtuale (nel quale tutti i pacchetti fra una data sorgente e una data destinazione seguiranno lo stesso percorso), esso deve

comunque preoccuparsi di determinare il percorso di un pacchetto. Questo è il compito del **protocollo di instradamento** (*routing protocol*) dello strato di rete.

Il nucleo di ciascun protocollo di instradamento è un algoritmo (l'**algoritmo di instradamento**) che determina il percorso di un pacchetto. Lo scopo di un algoritmo di instradamento è semplice: dato un gruppo di router, collegati da vari link, un algoritmo di instradamento trova un “buon” percorso dalla sorgente alla destinazione. Tipicamente, un “buon” percorso è quello che ha il “minimo costo”. L’astrazione grafica usata per formulare gli algoritmi di instradamento è illustrata in Figura 2.2.

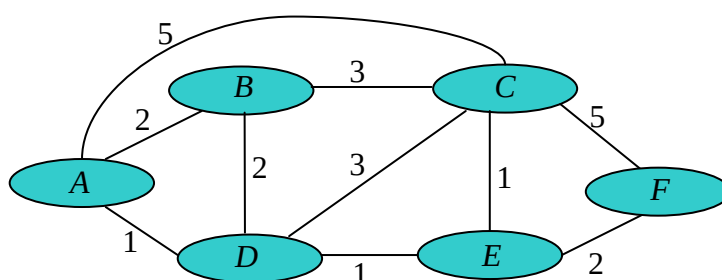


Figura 2.2: Modello astratto di una rete.

Nella figura, i nodi rappresentano i router (i punti in cui sono prese le decisioni sull’instradamento dei pacchetti) e le linee (“archi” nella terminologia della teoria dei grafi) che collegano questi nodi rappresentano i link fisici fra questi router. Un link ha anche un valore che rappresenta il “costo” della spedizione di un pacchetto attraverso il link. Il costo può riflettere il livello di congestione di quel link (per esempio, un link transoceanico può avere un costo più alto di un link terrestre che copre una breve distanza).

Data l’astrazione grafica, il problema di trovare il percorso a costo più basso da una sorgente a una destinazione richiede di determinare una serie di link in modo tale che:

- il primo link sul percorso sia collegato alla sorgente;
- l’ultimo link sul percorso sia collegato alla destinazione;
- per tutti gli i , i link i e $(i - 1)$ – esimo sul percorso siano collegati allo stesso nodo;
- per il **percorso di minor costo**, la somma dei costi di un link sul percorso è la minima tra quelle di tutti i percorsi possibili fra sorgente e destinazione. Si può osservare che, se i costi di tutti i link sono uguali, il percorso di minor costo coincide con il **percorso più corto** (cioè, il percorso che attraversa il minor numero di link fra sorgente e destinazione).

Nella Figura 2.2, per esempio, il percorso di minor costo fra il nodo *A* (sorgente) e il nodo *C* (destinazione) è lungo la rotta *ADEC*.

Genericamente, un modo per classificare gli algoritmi di instradamento è in funzione del fatto che siano globali o centralizzati:

- un **algoritmo di instradamento globale** (*global routing algorithm*) calcola il percorso di minor costo fra una sorgente e una destinazione usando conoscenze sulla rete complete, globali. Vale a dire, l'algoritmo prende come input la connettività fra tutti i nodi e tutti i costi dei link. Questo presuppone allora che l'algoritmo ottenga in qualche modo queste informazioni prima di eseguire effettivamente il calcolo. Il calcolo stesso può essere eseguito in un posto (un algoritmo di instradamento globale centralizzato) o replicato in più siti. Qui il fattore caratteristico principale, comunque, è che un algoritmo globale ha informazioni *complete* sulla connettività e sui costi dei link. In pratica, agli algoritmi con informazioni di stato globali spesso ci si riferisce come agli **algoritmi basati sullo stato dei link** (*link state algorithm*), poiché l'algoritmo deve essere a conoscenza del costo di ciascun link nella rete;
- in un **algoritmo di instradamento decentralizzato** (*decentralized routing algorithm*), il calcolo del percorso di minor costo è eseguito in modo interattivo e distribuito. Nessun nodo ha informazioni complete sul costo di tutti i link della rete. Piuttosto, ciascun nodo comincia con la conoscenza del costo dei link ad esso direttamente collegati. Allora, attraverso un processo interattivo di calcolo e scambio di informazioni con i nodi ad esso vicini (vale a dire, i nodi collegati all'altra estremità dei link cui esso è collegato), un nodo calcola gradualmente il percorso di minor costo verso una destinazione o un gruppo di destinazioni. Un tipico algoritmo di instradamento decentralizzato è l'**algoritmo distance vector**. Esso è detto algoritmo "distance vector" perché un nodo non può mai realmente conoscere un percorso completo dalla sorgente alla destinazione. Esso conosce solo il nodo vicino a cui deve inviare un pacchetto allo scopo di raggiungere una data destinazione lungo il percorso di minor costo, e il costo del percorso da se stesso alla destinazione.

2.1 Algoritmo di instradamento basato sullo stato del link

Per un algoritmo basato sullo stato dei link (*link state algorithm*), la topologia della rete e tutti i costi dei link sono noti; cioè, disponibili come input dell'algoritmo. In pratica, ciò si realizza con ciascun nodo che **trasmette** le identità e i costi dei link ad esso collegati a tutti i router nella rete. Questa **trasmissione dello stato dei link** (*link state broadcast*) può avvenire senza che i nodi

debbano conoscere inizialmente le identità di tutti gli altri nodi della rete. Un nodo richiede solo di conoscere l'identità e i costi di collegamento che sono direttamente collegati ad esso; conoscerà la topologia del resto della rete attraverso la ricezione delle trasmissioni dello stato dei link effettuate dagli altri nodi. Il risultato della trasmissione dello stato dei link di tutti i nodi è che tutti i nodi hanno un'identica e completa visione della rete. Ciascun nodo può allora far funzionare l'algoritmo dello stato dei link e calcolare lo stesso gruppo di percorsi di minor costo come qualsiasi altro ruolo.

L'algoritmo dello stato dei link più conosciuto è l'algoritmo di Dijkstra, dal nome del suo inventore. L'algoritmo di Dijkstra è iterativo e ha la proprietà che dopo la k – esima iterazione esso determina il percorso di minor costo per k nodi di destinazione, e fra tutti i percorsi di minor costo verso tutti i nodi di destinazione, questi k percorsi avranno i k costi minori.

Senza entrare nei dettagli dell'algoritmo, si può osservare che l'algoritmo basato sullo stato del link (LS), di cui Dijkstra ne rappresenta un esempio, consiste di un passo di inizializzazione seguito da un ciclo (*loop*). Il numero di loop eseguiti è uguale al numero dei nodi nella rete. Al termine l'algoritmo avrà calcolato i percorsi più corti dal nodo sorgente a tutti gli altri nodi nella rete e ha una complessità, nel peggiore dei casi, del tipo $O(n^2)$, dove n è il numero di nodi della rete.

Occorre considerare, in merito all'algoritmo LS, una patologia che può insorgere. La Figura 2.3 mostra una semplice topologia di rete dove il costo dei link è uguale al carico smaltito dai link; questa misura riflette, per esempio, il ritardo che si subisce. In questo esempio, i costi dei link non sono simmetrici, cioè il costo da A a B è uguale al costo da B ad A solo se il carico trasportato in entrambe le direzioni sul link AB è lo stesso. In questo esempio, il nodo D origina un'unità di traffico destinata ad A e il nodo C inserisce una quantità di traffico uguale ad e , anch'esso destinato ad A . L'instradamento iniziale è illustrato in Figura 2.3a) con i costi dei link corrispondenti al traffico trasportato.

Quando l'algoritmo LS passa alla fase successiva, il nodo C determina (basandosi sui costi dei link in Figura 2.3a) che il percorso in senso orario verso A ha costo 1, mentre quello in senso antiorario sempre verso A (che è stato usato) ha un costo di $1 + e$. Quindi il percorso da C verso A con il minor costo è quello in senso orario. Allo stesso modo, B determina che il percorso con minor costo verso A è anch'esso in senso orario, dando luogo ai costi mostrati nella Figura 2.3b. Quando l'algoritmo LS passa alla fase successiva, i nodi B , C , e D rilevano tutti un percorso a costo zero verso A in senso antiorario, e instradano tutti il loro traffico sul percorso in senso antiorario. Nella fase successiva di funzionamento dell'algoritmo LS, tutti i nodi B , C e D instraderanno il loro traffico in senso antiorario.

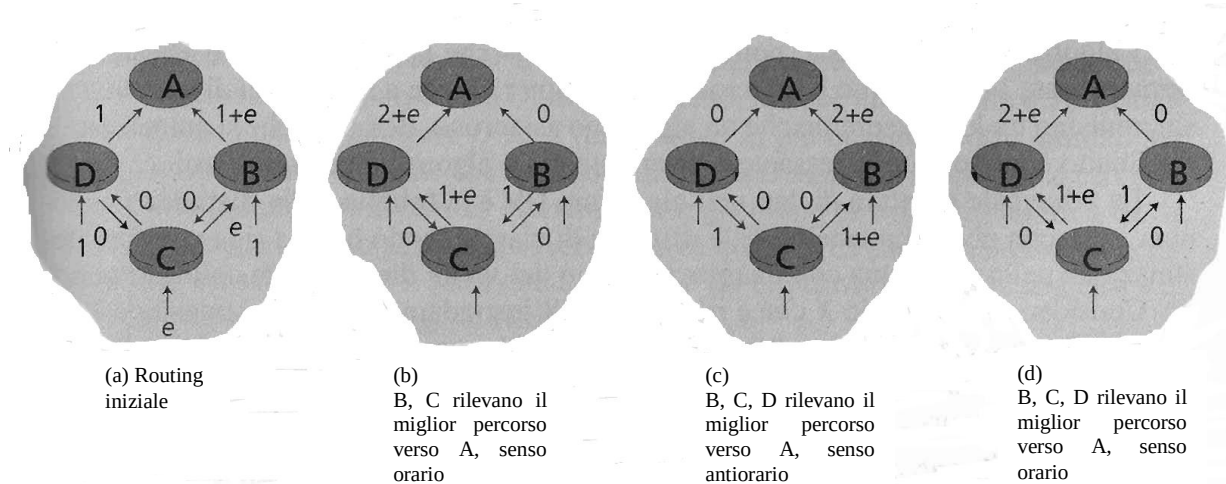


Figura 2.3: oscillazioni nell'instradamento.

In pratica, si creano dei cicli nel percorso dell'informazione, un fenomeno che si verifica in qualsiasi algoritmo di instradamento che usa una metrica di link basata sulla congestione o sul ritardo.

Una soluzione potrebbe essere di imporre che i costi dei link non dipendano dalla quantità di traffico trasportato, una soluzione inaccettabile perché un obiettivo dell'instradamento è di evitare l'utilizzo dei link fortemente congestionati (per esempio, caratterizzati da elevati ritardi).

Un'altra soluzione è di evitare che tutti i router mettano in funzione l'algoritmo LS nello stesso momento. Questa soluzione sembra più ragionevole, poiché possiamo sperare che sebbene l'algoritmo LS funzioni sui router con la stessa periodicità, l'istante di esecuzione dell'algoritmo non sarà lo stesso in tutti i nodi.

2.2 Algoritmo di instradamento distance vector

Mentre l'algoritmo LS è un algoritmo che usa l'informazione globale, l'algoritmo **distance vector** (**DV**, vettore delle distanze) è iterativo, asincrono e distribuito. Esso è distribuito perché ciascun nodo riceve alcune informazioni da uno o più "vicini" ai quali è direttamente collegato, esegue un calcolo, e può quindi distribuire i risultati del suo calcolo indietro ai suoi vicini. E' iterativo perché questo processo continua finché non viene più scambiata alcuna informazione tra i vicini (l'algoritmo si auto - arresta). L'algoritmo è asincrono perché non richiede a tutti i nodi di operare con sincronismo tra loro.

La principale struttura dati nell'algoritmo DV è la **tabella delle distanze** mantenuta a ciascun nodo, che permette una forma di comunicazione tra nodi vicini (*neighbour – to – neighbor*) che si svolge nell'algoritmo DV: ciascun nodo deve conoscere il costo di ciascuno dei percorsi di minor costo dei suoi vicini per ciascuna destinazione. Quindi, in qualunque momento un nodo calcola un nuovo costo minimo verso una destinazione, esso deve informare i suoi vicini di questo nuovo costo minimo. A partire dalla tabella delle distanze, è possibile una **tabella di instradamento** di un nodo (che indica quale link in uscita dovrebbe essere usato per inoltrare i pacchetti a una data destinazione e che, in sostanza, dà un'idea delle probabilità di instradare i pacchetti su un determinato link di uscita).

2.3 Algoritmi RA1 e RA2

Gli algoritmi di instradamento utilizzati nel modello fluidodinamico per reti di telecomunicazioni sono i seguenti:

RA1: I pacchetti dalle linee entranti vanno su quelle uscenti in base alla destinazione finale, massimizzando il numero di pacchetti solo sulle linee entranti.

RA2: I pacchetti dalle linee entranti vanno instradati su quelle uscenti al fine di massimizzare il numero di pacchetti sulle linee entranti ed uscenti.

La differenza principale nei due algoritmi è che mentre il primo è indifferente a possibili sovraccarichi della rete, e si comporta con abuso di notazione come uno “switch”, il secondo indirizza i pacchetti sulle linee uscenti tenendo in considerazione i carichi e dunque ridirezionando i pacchetti in modo tale da massimizzare il flusso sia sulle linee entranti che su quelle uscenti. Ancora con qualche abuso di notazione, potremmo dire che il secondo algoritmo si comporta come un “router”.

In pratica, l'algoritmo RA1 smista sulle linee di uscita non tenendo conto del congestionamento, a differenza dell'algoritmo RA2.

L'algoritmo RA1, già usato per i modelli di traffico stradale, prevede l'uso di una matrice di distribuzione del traffico, i cui coefficienti descrivono la percentuale di pacchetti che dalle linee entranti vengono indirizzati sulle linee uscenti. Per l'esistenza delle soluzioni ai problemi di Cauchy, ci si è ristretti al caso di nodi semplici con due linee entranti e due linee uscenti ma, a differenza del traffico stradale, è stato ottenuto un bound preciso sulla variazione totale della densità, grazie ad alcune ipotesi sulla funzione flusso, ed è stato poi possibile derivare l'esistenza delle soluzioni ai problemi di Cauchy, attraverso il metodo del wave-front tracking. Non è stato invece possibile provare la dipendenza continua dai dati.

La figura 2.4 mostra un nodo di una rete, composto da due linee entranti e due linee uscenti. In questo caso, la matrice di distribuzione del traffico è caratterizzata, essenzialmente, da due parametri, α , che indica la percentuale di pacchetti che dalla linea 1 viene instradata verso la linea 3, e β , che rappresenta la percentuale di pacchetti che dalla linea 2 viene direzionata verso la linea 3. Si noti che le probabilità complementari $1-\alpha$ e $1-\beta$ rappresentano, rispettivamente, le percentuali di pacchetti che si dirigono dalla linea 1 alla linea 4 e dalla linea 2 alla linea 4.

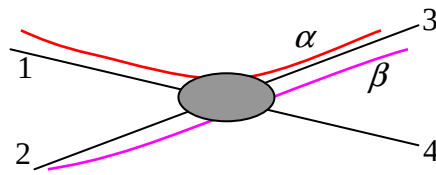


Figura 2.4: comportamento dell'algoritmo RA1 in un nodo della rete.

Per quanto riguarda il secondo algoritmo, non considerato per il traffico stradale in quanto il ridirezionamento non è previsto da un punto di vista modellistico (fenomeno che si verifica in casi eccezionali, quali forti congestioni o chiusura di strade), la dinamica alla giunzione (che comporta l'unicità delle soluzioni ai Problemi di Riemann) viene descritta mediante dei parametri di priorità, che descrivono le priorità tra i flussi di pacchetti sulle linee entranti, e coefficienti di distribuzione che hanno lo stesso significato di quelli della matrice di distribuzione del traffico.

L'algoritmo RA2, a differenza dell'RA1, ha un funzionamento regolato sul massimo flusso in ingresso ad un nodo e il massimo flusso in uscita dallo stesso nodo.

In particolare, se il massimo flusso in arrivo al nodo (proveniente dalle linee di ingresso) è minore del massimo flusso che può essere smistato sulle linee di uscita, allora tutta l'informazione che giunge al nodo può essere smistata in uscita. Quindi, vi è bisogno di alcuni coefficienti di distribuzione che indichino le percentuali di flusso di ingresso che competono ad ogni linea di uscita. La situazione è illustrata in figura 2.5 dove, in particolare α e $1-\alpha$ indicano, rispettivamente, la percentuale di flusso che dal nodo va verso la linea di uscita 1 e verso la linea di uscita 2, rispettivamente.

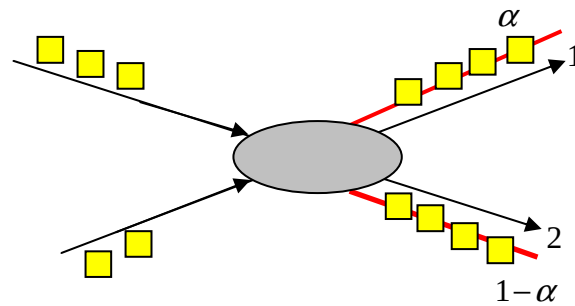


Figura 2.5:

Algoritmo RA2 nel caso dell'uso dei coefficienti di distribuzione. I quadratini indicano una misura della capacità massima di ogni linea. Le linee rosse indicano le direzioni dei flussi di traffico in uscita dal nodo.

Se il massimo flusso in arrivo al nodo (proveniente dalle linee di ingresso) è maggiore del massimo flusso che può essere instradato sulle linee di uscita, allora non è possibile smistare sulle linee di uscita tutta l'informazione che giunge al nodo. Quindi, vi è bisogno di alcuni parametri di priorità che indichino quali sono le percentuali dei flussi sulle linee di ingresso che massimizzano i flussi smistati sulle linee di uscita. La situazione è illustrata in figura 2.6 dove, in particolare p e $1-p$ indicano, rispettivamente, le percentuali di flusso che dalla linea 1 e dalla linea 2, rispettivamente, vanno verso il nodo.

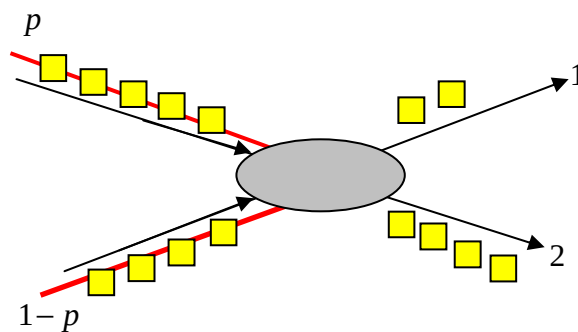


Figura 2.6:

Algoritmo RA2 nel caso dell'uso dei parametri di priorità. I quadratini indicano una misura della capacità massima di ogni linea. Le linee rosse indicano le direzioni dei flussi di traffico in ingresso al nodo.

Il vantaggio dell'algoritmo RA2 è che la variazione del flusso in un nodo è conservata per interazioni di onde provenienti da linee di trasmissione, il che permette di ottenere stime sulla variazione totale delle densità, e di costruire una soluzione entropica ai problemi di Cauchy. Inoltre, utilizzando le tecniche dei vettori tangenti, si è dimostrata l'unicità e la dipendenza Lipschitziana per soluzioni sull'intera rete. Questi ultimi risultati si differenziano da quelli ottenuti per il caso di traffico stradale. Il motivo principale risiede nella possibilità di re-indirizzamento che modella il

comportamento di routing di certe reti di telecomunicazione. Il risultato di dipendenza continua dei dati mostra che la scelta di un algoritmo di tipo “router” implica la stabilità delle soluzioni, rispetto a perturbazioni dei dati, contro l’instabilità ottenuta con uno di tipo “switch”.

In realtà gli algoritmi RA1 e RA2 possono essere ricondotti a quelli già descritti nella sezione 2.

In termini macroscopici il problema dell’instradamento può essere formulato matematicamente come un problema di ottimizzazione vincolata, mediante una matrice di distribuzione del traffico che contiene dei coefficienti statistici (parametri di instradamento) che indicano come il traffico, ad un router, da un dato link di ingresso i viene smistato verso un link di uscita j . I vincoli di cui si deve tener conto sono i seguenti:

- la somma dei flussi fra tutte le coppie sorgente-destinazione che passano attraverso il link m devono essere inferiori alla capacità del link m (vincolo sul massimo flusso per ogni link);
- la quantità di traffico che entra in ciascun router r (sia da altri router, sia direttamente dall’host collegato a quel router) deve essere uguale alla quantità di traffico che lascia il router, o attraverso un link in uscita da r o attraverso l’host collegato direttamente a quel router (vincolo di conservazione del flusso).

Il problema di ottimizzazione è allora trovare un insieme di flussi che convergono ai router in maniera tale da soddisfare i vincoli presentati. La soluzione di questo problema di ottimizzazione definisce allora l’instradamento usato nella rete.

In particolare, gli algoritmi di instradamento possono essere di due tipi, in questo caso:

- un algoritmo (Routing algorithm 1, RA1) che decide che, ad un determinato nodo della rete, i flussi debbano essere smistati con una certa probabilità verso i link di uscita (utilizzando così una matrice di distribuzione del traffico); tale algoritmo, lavorando su tabelle di instradamento, è del tipo “distance vector”.
- un algoritmo (Routing algorithm 2, RA2) che decide come ridirezionare i pacchetti in accordo alla congestione dei link. In particolare, quindi, l’algoritmo RA2, proprio perché lavora sulle condizioni di congestione sui link (e quindi dei costi sui link) è di tipo “link – state”.

3 Definizione degli algoritmi di instradamento nel modello fluidodinamico

In questa sezione, si vogliono dettagliare gli algoritmi di instradamento (RA1) e (RA2) per il modello di reti di telecomunicazioni con sorgenti e destinazioni. La sezione è organizzata nel seguente modo: nel paragrafo 3.1, si daranno delle definizioni di base; nel paragrafo 3.2, si discuterà delle dinamiche relative alle linee di trasmissione; nel paragrafo 3.3, si descriveranno i fenomeni associati alle dinamiche dei pacchetti sulla rete; nel paragrafo 3.4, si discuterà delle funzioni di tipo – traffico; nel paragrafo 3.5, si descriveranno i Riemann Solver che costituiscono le soluzioni dei Problemi di Riemann sulle linee entranti ed uscenti da un determinato nodo della rete. Si osservi che i Riemann Solver sono diversi a seconda dell'algoritmo di instradamento scelto.

3.1 Nozioni di base

Consideriamo una rete di telecomunicazioni, che è un insieme finito di linee di trasmissione unite da nodi, alcuni dei quali sono sorgenti e destinazioni. Formalmente, introduciamo la seguente definizione:

Definizione. Una rete di telecomunicazioni è data da una 7 - tupla $(\mathbf{N}, \mathbf{I}, \mathbf{F}, \mathbf{J}, \mathbf{S}, \mathbf{D}, \mathbf{R})$ dove

- $\mathbf{N}$ è la cardinalità della rete, ovvero il numero di linee della rete;
- $\mathbf{I}$ è la collezione delle linee, modellate da intervalli $I_i = [a_i, b_i] \subseteq \mathbb{R}$, $i = 1, \dots, N$;
- $\mathbf{F}$ è la collezione delle funzioni flusso $f_i : [0, \rho_i^{\max}] \mapsto \mathbb{R}$, $i = 1, \dots, N$;
- $\mathbf{J}$ è una collezione di sottoinsiemi di $\{\pm 1, \dots, \pm N\}$ che rappresentano i nodi. Se $j \in J \in \mathbf{J}$, allora la linea di trasmissione $I_{|j|}$ è una linea entrante per la giunzione J (al punto b_i) se $j > 0$ ed è una linea uscente (al punto a_i) se $j < 0$. Per ogni giunzione $J \in \mathbf{J}$, indichiamo con $\text{Inc}(J)$ l'insieme delle linee entranti, che sono le linee I_i tali che $i \in J$ e $i > 0$, mentre con $\text{Out}(J)$ l'insieme delle linee uscenti, che sono le linee I_i tali che $-i \in J$ e $i > 0$.
Assumiamo che ogni linea sia entrante per almeno un nodo e uscente per almeno un nodo;
- $\mathbf{S}$ è il sottoinsieme di $\{1, \dots, N\}$ che rappresenta le linee che cominciano da sorgenti di traffico. Quindi, $j \in \mathbf{S}$ se e solo se j non è uscente per qualche nodo. Assumiamo che $\mathbf{S} \neq \emptyset$;
- $\mathbf{D}$ è il sottoinsieme di $\{1, \dots, N\}$ che rappresenta le linee che portano alle destinazioni di traffico. Così, $j \in \mathbf{D}$ se e solo se j non è entrante per qualche nodo. Assumiamo che $\mathbf{D} \neq \emptyset$;
- $\mathbf{R}$ è una collezione finita di funzioni r_j . Per ogni J , $r_j(i, s, d)$ indica la direzione di uscita

del traffico che, cominciato alla sorgente s , ha d come destinazione finale, e che ha raggiunto J dalla strada entrante i (considereremo anche il caso in cui r_j è a più valori).

Di solito, si assume che la rete sia connessa, condizione non strettamente necessaria.

3.2 Dinamiche sulle linee

Per la descrizione delle dinamiche delle densità dei pacchetti lungo le linee si fanno le seguenti ipotesi:

(H1) le linee sono composte da processori consecutivi N_k , che ricevono e mandano pacchetti. Il numero dei pacchetti nel nodo N_k viene indicato con $R_k \in [0, R_{\max}]$;

(H2) esistono due scale temporali: Δt_0 , che rappresenta il tempo di viaggio fisico di un singolo pacchetto da un nodo ad un altro (assunta l'indipendenza dal nodo per semplicità); T , che rappresenta il tempo di processamento, durante il quale ogni processore cerca di effettuare la trasmissione di un dato pacchetto;

(H3) ogni processore N_k cerca di spedire tutti i pacchetti R_k allo stesso tempo. I pacchetti vengono persi secondo una funzione di probabilità $p : [0, R_{\max}] \rightarrow [0, 1]$, determinata in R_{k+1} , e i pacchetti persi vengono rispediti ancora in un intervallo temporale di ampiezza T .

Lo scopo è determinare i flussi sulla rete. Dal momento che la velocità di trasmissione di pacchetto sulla linea può essere assunta costante, è possibile calcolare una funzione di velocità media e quindi una funzione di flusso medio.

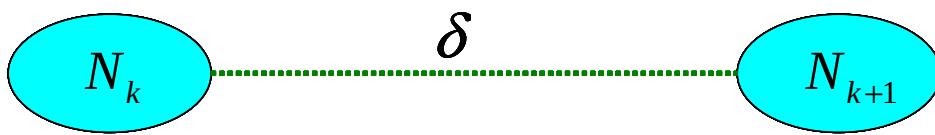


Figura 3.1: Linea di trasmissione tra nodi consecutivi.

Focalizziamo l'attenzione su due nodi consecutivi, N_k e N_{k+1} , assumiamo una situazione statica, per esempio R_k e R_{k+1} sono costanti, e sia δ la distanza tra i nodi. Durante un intervallo di processamento pari a T , accadono i seguenti fenomeni. Tutti i pacchetti R_k vengono spediti una prima volta: $(1 - p(R_{k+1}))R_k$ sono spediti con successo e $p(R_{k+1})R_k$ vengono persi. Al secondo tentativo, dei pacchetti persi $p(R_{k+1})R_k$, $(1 - p(R_{k+1}))p(R_{k+1})R_k$ vengono spediti con successo e $p^2(R_{k+1})R_k$ vengono persi, e così via.

Indichiamo con Δt_{av} la velocità di trasmissione media dei pacchetti, con $\bar{v} = \frac{\delta}{\Delta t_0}$ la velocità dei pacchetti senza perdite e con $v = \frac{\delta}{\Delta t_{av}}$ la velocità media dei pacchetti. Allora, possiamo calcolare:

$$\Delta t_{av} = \sum_{n=1}^M n \Delta t_0 (1 - p(R_{k+1})) p^{n-1}(R_{k+1}),$$

dove $M = [T / \Delta t_0]$ (qui $[\cdot]$ indica la funzione parte intera superiore) rappresenta il numero di tentativi associati all'invio di un pacchetto. Facciamo, inoltre, la seguente assunzione:

(H4) Il numero di pacchetti non trasmessi per un intero slot temporale è trascurabile.

Tale ipotesi corrisponde ad assumere che $\Delta t_0 \ll T$ o, equivalentemente, che $M \sim +\infty$. Per $M = +\infty$, si ha:

$$\Delta t_{av} = \sum_{n=1}^{+\infty} n \Delta t_0 (1 - p(R_{k+1})) p^{n-1}(R_{k+1}) = \frac{\Delta t_0}{1 - p(R_{k+1})},$$

e

$$v = \frac{\delta}{\Delta t_{av}} = \frac{\delta}{\Delta t_0} (1 - p(R_{k+1})) = \bar{v} (1 - p(R_{k+1})). \quad (3.1)$$

Sia ora ρ la densità media e $\rho_{\max}$ il suo massimo. Possiamo interpretare la funzione p come funzione di ρ e, usando l'espressione (3.1), determinare la corrispondente funzione di flusso, data dal prodotto tra la densità media e il flusso medio. È ragionevole assumere che la funzione relativa alla probabilità di perdita sia zero per intervalli che si trovano vicini allo zero. Questo significa che, a basse densità, non si ha perdita di pacchetto. La funzione p dovrebbe aumentare raggiungendo il valore 1 in corrispondenza della massima densità, situazione corrispondente ad una linea bloccata. Una possibile scelta per la funzione di flusso è la seguente:

$$p(\rho) = \begin{cases} 0, & 0 \leq \rho \leq \sigma, \\ \frac{\rho_{\max}(\rho - \sigma)}{\rho(\rho_{\max} - \sigma)}, & \sigma \leq \rho \leq \rho_{\max}, \end{cases}$$

da cui segue che

$$f(\rho) = \begin{cases} \bar{v} \rho, & 0 \leq \rho \leq \sigma, \\ \frac{\bar{v} \sigma (\rho_{\max} - \rho)}{\rho_{\max} - \sigma}, & \sigma \leq \rho \leq \rho_{\max}. \end{cases} \quad (3.2)$$

Ponendo, per semplicità, $\rho_{\max} = 1$ e $\sigma = \frac{1}{2}$, si ha la semplice funzione “a tendina”.

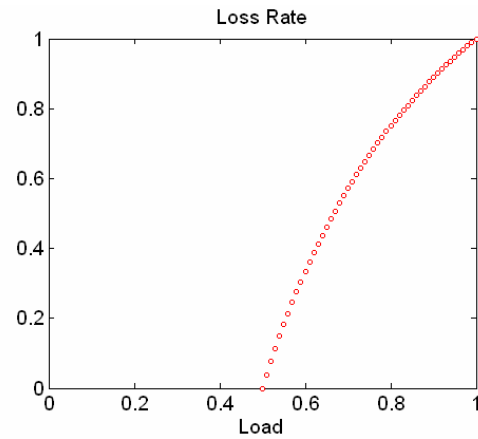


Figura 3.2: andamento della probabilità di perdita.

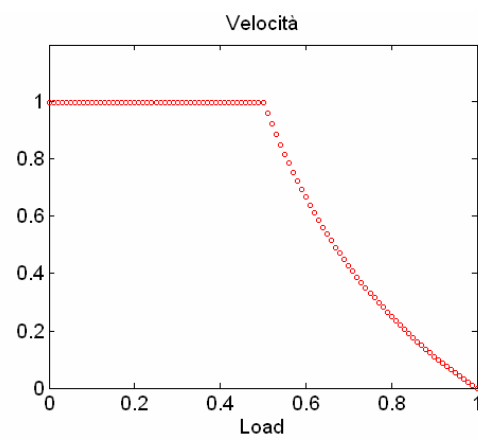


Figura 3.3: andamento della funzione velocità.

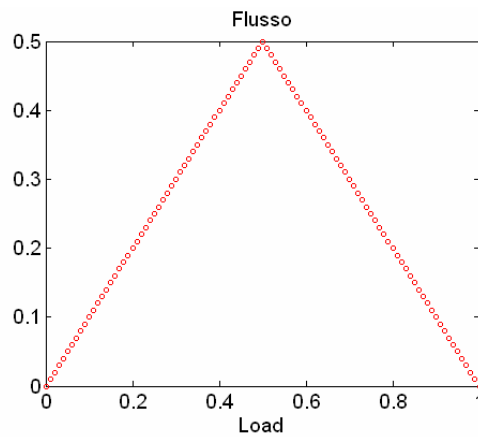


Figura 3.4: andamento della funzione flusso (a tendina).

Al fine di semplificare ulteriormente la trattazione assumiamo le seguenti ipotesi:

- (F)** Posto $\rho_{\max} = 1$, su ogni linea il flusso $f_i : [0, 1] \rightarrow R$ è concavo, $f(0) = f(1) = 0$, ed esiste un unico punto di massimo $\sigma \in]0, 1[$.

Si noti che il flusso di figura 3.4 o, più generalmente, il flusso dato dalla (3.2), soddisfa l'assunzione (F).

3.3 Dinamiche sulla rete

Su ogni linea di trasmissione I_i , consideriamo l'equazione di evoluzione

$$\partial_t \rho_i + \partial_x f_i(\rho_i) = 0, \quad (3.3)$$

dove usiamo l'assunzione (F). Quindi, l'evoluzione del carico della rete viene descritta da un set finito di funzioni

$$\rho_i : [0, +\infty[\times I_i \mapsto [0, \rho_i^{\max}].$$

Su ogni linea di trasmissione I_i , richiediamo che ρ_i sia una soluzione debole entropica della (3.3), ovvero per ogni funzione test $\varphi : [0, +\infty[\times I_i \rightarrow \mathbb{R}$, regolare positiva e a supporto compatto su $]0, +\infty[\times]a_i, b_i[$, deve aversi

$$\int_0^{+\infty} \int_{a_i}^{b_i} \left(\rho_i \frac{\partial \varphi}{\partial t} + f_i(\rho_i) \frac{\partial \varphi}{\partial x} \right) dx dt = 0,$$

e per ogni $k \in \mathbb{R}$ e ogni $\tilde{\varphi} : [0, +\infty[\times I_i \rightarrow \mathbb{R}$, regolare positiva e a supporto compatto su $]0, +\infty[\times]a_i, b_i[$, deve aversi

$$\int_0^{+\infty} \int_{a_i}^{b_i} \left(|\rho_i - k| \frac{\partial \tilde{\varphi}}{\partial t} + \text{sgn}(\rho_i - k) (f_i(\rho_i) - f_i(k)) \frac{\partial \tilde{\varphi}}{\partial x} \right) dx dt \geq 0.$$

Per ogni $i \in \mathbf{S}$ (risp. $i \in \mathbf{D}$), bisogna introdurre funzione di inflow (risp. outflow), ovvero si considerano funzioni misurabili $\psi_i : [0, +\infty[\rightarrow [0, \rho_i^{\max}]$ e le corrispondenti funzioni ρ_i devono verificare la condizione al bordo $\rho_i(t, a_i) = \psi_i(t)$ (risp. $\rho_i(t, b_i) = \psi_i(t)$) nel senso di [4].

Inoltre, per ogni linea I_i , definiamo una funzione tipo – traffico π_i , che misura la porzione di densità totale proveniente da ogni sorgente e che viaggia verso ogni destinazione:

Definizione. Una funzione tipo – traffico su una linea I_i è una funzione:

$$\pi_i : [0, \infty[\times]a_i, b_i[\times \mathbf{S} \times \mathbf{D} \mapsto [0, 1]$$

tale che, per ogni $t \in [0, \infty[$ e $x \in]a_i, b_i[$,

$$\sum_{s \in \mathbf{S}, d \in \mathbf{D}} \pi_i(t, x, s, d) = 1.$$

In altre parole, $\pi_i(t, x, s, d)$ specifica la frazione di densità $\rho_i(t, x)$ che ha avuto inizio dalla sorgente s e si sta propagando verso la destinazione finale d .

Abbiamo assunto, nel modello discreto, che ai nodi si utilizzi una politica FIFO. Di conseguenza, è naturale che la velocità media, ottenuta nella procedura di limite, risulta indipendente dalle sorgenti dei pacchetti e dalle loro destinazioni finali. In altre parole, facciamo la seguente ipotesi:

(H5) Su ogni linea I_i , la velocità media dei pacchetti dipende solo dal valore della densità ρ_i e non dai valori della funzione tipo – traffico π_i .

Come conseguenza dell'ipotesi (H5), si ha che se $x(t)$ indica la traiettoria di un pacchetto all'interno della linea I_i , otteniamo che:

$$\pi_i(t, x(t), s, d) = \text{const.} \quad (3.4)$$

Infatti, consideriamo i pacchetti che, al tempo t , sono in posizione $x(t)$. Tali pacchetti hanno la stessa velocità, per la (H5), così le loro traiettorie coincidono, indipendentemente dalle sorgenti e dalle destinazioni. In altre parole, al tempo $t' > t$, tutti i pacchetti saranno in posizione $x(t')$. Allora, le frazioni di densità, espresse da π , sono le stesse a $(t, x(t))$ e a $(t', x(t'))$. Considerando il differenziale totale rispetto al tempo della (4), deduciamo la seguente equazione semilineare:

$$\partial_t \pi_i(t, x, s, d) + \partial_x \pi_i(t, x, s, d) \cdot v_i(\rho_i(t, x)) = 0. \quad (3.5)$$

Questa equazione è accoppiata all'equazione (3.3) su ogni linea I_i . Più precisamente, l'equazione (3.5) dipende dalla soluzione della (3.4), mentre alle giunzioni il valore di π_i determinerà la distribuzione del traffico sulle linee uscenti, come spiegato in precedenza.

Per semplicità e senza perdita di generalità, assumiamo, d'ora in poi, che i flussi f_i siano tutti uguali ed indichiamoli con f . Quindi, il modello per una singola linea di trasmissione consiste di un sistema di equazioni:

$$\begin{cases} \rho_t + f(\rho)_x = 0, \\ \pi_t + \pi_x \cdot v(\rho) = 0. \end{cases}$$

Per trattare l'evoluzione alla giunzione, introduciamo alcune notazioni. Consideriamo una giunzione J con n linee di trasmissione entranti, indicate con $I_1, \dots, I_n$, e m linee di trasmissione uscenti, $I_{n+1}, \dots, I_{n+m}$. Una soluzione debole per J è una collezione di funzioni $\rho_l : [0, +\infty[\times I_l \rightarrow \mathbb{R}$, $l = 1, \dots, n+m$, tali che

$$\sum_{l=1}^{n+m} \left(\int_0^{+\infty} \int_{a_l}^{b_l} \left(\rho_l \frac{\partial \varphi_l}{\partial t} + f(\rho_l) \frac{\partial \varphi_l}{\partial x} \right) dx dt \right) = 0,$$

per ogni φ_l , $l=1, \dots, n+m$, regolare e con supporto compatto in $]0, +\infty[\times]a_l, b_l]$ per $l=1, \dots, n$ (linee di trasmissione entranti) e in $]0, +\infty[\times]a_l, b_l[$ per $l=n+1, \dots, n+m$ (linee di trasmissione uscenti), che sono anche regolari in corrispondenza della giunzione, ovvero:

$$\varphi_i(\cdot, b_i) = \varphi_j(\cdot, a_j), \quad \frac{\partial \varphi_i}{\partial x}(\cdot, b_i) = \frac{\partial \varphi_j}{\partial x}(\cdot, a_j), \quad i=1, \dots, n, j=n+1, \dots, n+m.$$

Osservazione Sia $\rho = (\rho_1, \dots, \rho_{n+m})$ una soluzione debole alla giunzione J tale che ogni $x \rightarrow \rho_i(t, x)$ abbia variazione limitata. Possiamo dedurre che ρ soddisfa la condizione di Rankine Hugoniot a J , ovvero che

$$\sum_{i=1}^n f(\rho_i(t, b_i-)) = \sum_{j=n+1}^{n+m} f(\rho_j(t, a_j+)),$$

per quasi ogni $t > 0$.

Per una legge di conservazione scalare, un problema di Riemann è un problema di Cauchy per un dato iniziale di tipo Heavyside, che è una funzione costante a tratti con una sola discontinuità. Si vogliono determinare soluzioni del tipo $\rho(t, x) = \phi(\frac{x}{t})$, formate da onde semplici, che sono il punto di partenza per costruire soluzioni al problema di Cauchy mediante l'algoritmo del wave – front tracking. Queste soluzioni sono formate da onde continue, chiamate “rarefazioni”, e da discontinuità viaggianti, chiamate “shocks”. La velocità di tali onde è strettamente legata ai valori della funzione f' (si veda [5], [7], [18]).

Analogamente, definiamo “problema di Riemann” per una giunzione il problema di Cauchy per dati iniziali $\rho_{1,0}, \dots, \rho_{n+m,0} \in [0,1]$ e $\pi_1^{s,d}, \dots, \pi_{n+m}^{s,d} \in [0,1]$ costanti per ogni linea di trasmissione.

Definizione. Un Riemann Solver (RS) per la giunzione J è una mappa che associa a dei dati di Riemann $\rho_0 = (\rho_{1,0}, \dots, \rho_{n+m,0})$ e $\Pi_0 = (\pi_{1,0}, \dots, \pi_{n+m,0})$ in J i vettori $\hat{\rho} = (\hat{\rho}_1, \dots, \hat{\rho}_{n+m})$ e $\hat{\pi} = (\hat{\pi}_1, \dots, \hat{\pi}_{n+m})$ tali che la soluzione su una linea di trasmissione entrante I_i , $i=1, \dots, n$, è data dall'onda $(\rho_{i,0}, \hat{\rho}_i)$, e su una linea uscente I_j , $j=n+1, \dots, n+m$, dalle onde $(\hat{\rho}_j, \rho_{j,0})$ e $(\hat{\pi}_j, \bar{\pi}_{j,0})$. Si richiede la condizione di consistenza:

$$RS(RS(\rho_0, \Pi_0)) = RS(\rho_0, \Pi_0).$$

Definiremo un RS alla giunzione in seguito. Una volta che è stato scelto un Riemann solver ed è stata ottenuta la soluzione del problema di Riemann, possiamo definire delle soluzioni ammissibili alle giunzioni.

Definizione. Assumiamo che sia assegnato un Riemann solver RS . Sia $\rho = (\rho_1, \dots, \rho_{n+m})$ tale che $\rho_i(t, \cdot)$ sia a variazione limitata per ogni $t \geq 0$. Allora, ρ è una soluzione debole ammissibile di (3) associata al RS alla giunzione J se e solo se valgono le seguenti proprietà:

(A) ρ è una soluzione debole alla giunzione J ;

(B) Π è una soluzione debole alla giunzione J ;

(C) per quasi ogni t , ponendo

$$\begin{aligned}\rho_J(t) &= (\rho_1(\cdot, b_1-), \dots, \rho_n(\cdot, b_n-), \rho_{n+1}(\cdot, a_{n+1}+), \dots, \rho_{n+m}(\cdot, a_{n+m}+)), \\ \Pi_J(t) &= (\pi_1(\cdot, b_1-), \dots, \pi_n(\cdot, b_n-), \pi_{n+1}(\cdot, a_{n+1}+), \dots, \pi_{n+m}(\cdot, a_{n+m}+)),\end{aligned}$$

abbiamo che

$$RS(\rho_J(t), \Pi_J(t)) = (\rho_J(t), \Pi_J(t)).$$

Specifichiamo cosa si intende per soluzione di una rete ammissibile (vedi [11]).

Definizione. Si consideri una rete ammissibile $(\mathbf{N}, \mathbf{I}, \mathbf{F}, \mathbf{J}, \mathbf{S}, \mathbf{D}, \mathbf{R})$. Un set di condizioni iniziali e al bordo (brevemente IBC) è dato assegnando delle funzioni misurabili $\bar{\rho}_i : I_i \mapsto [0, \rho_i^{\max}]$, $\bar{\pi}_i : [a_i, b_i] \times \mathbf{S} \times \mathbf{D} \mapsto [0, 1], i = 1, \dots, N$ e delle funzioni misurabili $\psi_i : [0, +\infty[\mapsto [0, \rho_i^{\max}]$, $i \in \mathbf{S} \cup \mathbf{D}$, e $\theta_{i,j} : [0, +\infty[\mapsto [0, 1], i \in \mathbf{S}, j \in \mathbf{D}$ con la proprietà che $\sum_j \theta_{i,j}(t) = 1$.

Definizione. Si consideri una rete ammissibile $(\mathbf{N}, \mathbf{I}, \mathbf{F}, \mathbf{J}, \mathbf{S}, \mathbf{D}, \mathbf{R})$ e un set di IBC. Un set di funzioni $\rho = (\rho_1, \dots, \rho_N)$ con $\rho_i : [0, +\infty[\times I_i \mapsto [0, \rho_i^{\max}]$ continue da $[0, +\infty[$ in L^1 , e $\Pi = (\pi_1, \dots, \pi_N)$ con $\pi_i : [0, +\infty[\times I_i \times \mathbf{S} \times \mathbf{D} \mapsto [0, 1]$, continue da $[0, +\infty[$ in L^1 , per ogni $s \in \mathbf{S}, d \in \mathbf{D}$, è una soluzione ammissibile se vale la seguente condizione: ogni ρ_i è una soluzione debole entropica della (3.3) su I_i , $\rho_i(0, x) = \bar{\rho}_i(x)$ per quasi ogni $x \in [a_i, b_i]$, $\rho_i(t, a_i) = \psi_i(t)$ se $i \in \mathbf{S}$ e $\rho_i(t, b_i) = \psi_i(t)$ se $i \in \mathbf{D}$ nel senso di [4]. Ogni π_i è una soluzione debole dell'equazione (4), $\pi_i(0, x) = \bar{\pi}_i(x)$ per quasi ogni $x \in [a_i, b_i]$ e per ogni $i \in \mathbf{S}$, $j \in \mathbf{D}$, $\pi_i^{i,j}(t, a_i) = \theta_{i,j}$ nel senso di [4]. Infine, ad ogni giunzione, (ρ, Π) è una soluzione debole ed è una soluzione debole ammissibile nel caso di variazione limitata.

3.4 Distribuzione di traffico alle giunzioni

Si consideri una giunzione J in cui ci siano n linee di trasmissione con traffico entrante e m linee di trasmissione con traffico uscente. Indichiamo con $\rho_i(t, x)$, $i = 1, \dots, n$ e $\rho_j(t, x)$, $j = n+1, \dots, n+m$ le densità di traffico, rispettivamente, sulle linee di trasmissione entranti e su quelle uscenti, e con $(\rho_{1,0}, \dots, \rho_{n+m,0})$ il dato iniziale.

Definiamo $\gamma_i^{\max}$ e $\gamma_j^{\max}$ nel seguente modo:

$$\gamma_i^{\max} = \begin{cases} f(\rho_{i,0}), & \text{se } \rho_{i,0} \in [0, \sigma], \\ f(\sigma), & \text{se } \rho_{i,0} \in]\sigma, 1], \end{cases} \quad i = 1, \dots, n, \quad (3.6)$$

e

$$\gamma_j^{\max} = \begin{cases} f(\sigma), & \text{se } \rho_{j,0} \in [0, \sigma], \\ f(\rho_{j,0}), & \text{se } \rho_{j,0} \in]\sigma, 1], \end{cases} \quad j = n+1, \dots, n+m. \quad (3.7)$$

Le quantità $\gamma_i^{\max}$ e $\gamma_j^{\max}$ rappresentano il massimo flusso che può essere ottenuto da una singola onda soluzione su ciascuna linea di trasmissione. Infine, indichiamo con Ω_i e Ω_j le quantità:

$$\begin{aligned} \Omega_i &= [0, \gamma_i^{\max}], \quad i = 1, \dots, n, \\ \Omega_j &= [0, \gamma_j^{\max}], \quad j = n+1, \dots, n+m, \end{aligned}$$

e con $\hat{\gamma}_{inc}$ e $\hat{\gamma}_{out}$ le quantità $\hat{\gamma}_{inc} = (f(\hat{\rho}_1), \dots, f(\hat{\rho}_n))$, $\hat{\gamma}_{out} = (f(\hat{\rho}_{n+1}), \dots, f(\hat{\rho}_{n+m}))$, dove $\hat{\rho} = (\hat{\rho}_1, \dots, \hat{\rho}_{n+m})$ è la soluzione del problema di Riemann alla giunzione.

Discutiamo ora alcune possibili scelte per la funzione di distribuzione del traffico:

- 1) $r_j : Inc(J) \times S \times D \rightarrow Out(J)$;
- 2) $r_j : Inc(J) \times S \times D \hookrightarrow Out(J)$, ovvero r_j è una multifunzione.

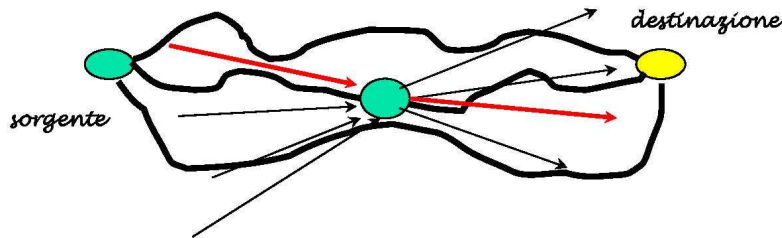


Figura 3.5: interpretazione grafica della funzione di distribuzione di traffico.

Se r_j è di tipo 1), allora ogni pacchetto ha un percorso deterministico, cioè alla giunzione J il traffico, che è iniziato alla sorgente s e ha d come destinazione finale e che proviene dalla linea di trasmissione i , viene indirizzato verso un'assegnata linea j ($r_j(i, s, d) = j$).

Invece, se r_j è di tipo 2), alla giunzione J il traffico con sorgente s e destinazione d , proveniente da una linea i , viene indirizzato su ogni linea $I_j \in Out(J)$ o su qualche linea $I_j \in Out(J)$. Possiamo definire $r_j(i, s, d)$ in due diversi modi:

$$2a) \quad \begin{aligned} r_j &: Inc(J) \times S \times D \rightarrow Out(J), \\ r_j(i, s, d) &\subseteq Out(J); \end{aligned}$$

$$2b) \quad r_j : Inc(J) \times S \times D \hookrightarrow [0,1]^{Out(J)},$$

$$r_j(i, s, d) = (\alpha_J^{i,s,d,n+1}, \dots, \alpha_J^{i,s,d,n+m}) \quad \text{con} \quad 0 \leq \alpha_J^{i,s,d,j} \leq 1, \quad j \in \{n+1, \dots, n+m\}, \quad \sum_{j=n+1}^{n+m} \alpha_J^{i,s,d,j} = 1.$$

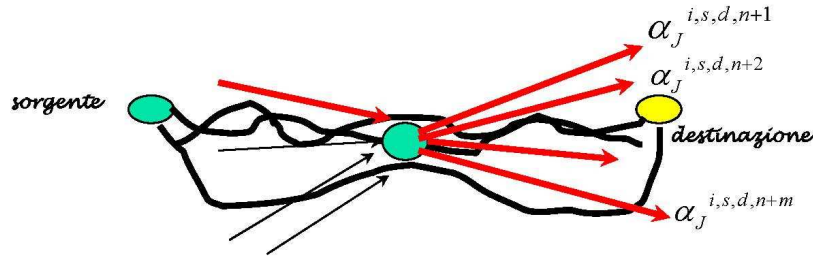


Figura 3.6: interpretazione grafica della funzione di distribuzione del tipo 2b).

Nel caso 2a), dobbiamo specificare in che modo il traffico alla giunzione J viene direzionato sulle linee di uscita.

La definizione 2b) significa che, alla giunzione J , il traffico con sorgente s e destinazione d , proveniente dalla linea I_i , viene indirizzato sulla linea di uscita j , $j = n+1, \dots, n+m$ con probabilità $\alpha_J^{i,s,d,j}$.

Definizione. Una matrice di distribuzione è una matrice

$$A \doteq \{\alpha_{j,i}\}_{j=n+1, \dots, n+m, i=1, \dots, n} \in \mathbb{R}^{m \times n}$$

tale che

$$0 < \alpha_{j,i} < 1, \quad \sum_{j=n+1}^{n+m} \alpha_{j,i} = 1,$$

per ogni $i = 1, \dots, n$ e $j = n+1, \dots, n+m$, dove $\alpha_{j,i}$ è la percentuale di pacchetti che arrivano dalla linea di trasmissione entrante i e che si dirigono verso la linea di trasmissione uscente j .

Nel caso 1) possiamo definire la matrice A nel seguente modo. Si fissi un tempo t e assumiamo che per ogni $i \in \text{Inc}(J)$, $s \in \mathbf{S}$, e $d \in \mathbf{D}$, $\pi_i(t, \cdot, s, d)$ ammetta limite alla giunzione J , ovvero limite sinistro a b_i . Per $i \in \{1, \dots, n\}$, $j \in \{n+1, \dots, n+m\}$, poniamo

$$\alpha_{j,i} = \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ r_j(i, s, d) = j}} \pi_i(t, b_i-, s, d).$$

Affinché i flussi $f_i(\rho_i)$ siano consistenti con le funzioni tipo – traffico, devono soddisfare la seguente relazione:

$$f_j(\rho_j(\square, a_j+)) = \sum_{i=1}^n \alpha_{j,i} f_i(\rho_i(\square, b_i-))$$

per ogni $j = n+1, \dots, n+m$.

Analizziamo come definire la matrice A nel caso 2a). Possiamo assegnare $\varphi(i, s, d) \in r_j(i, s, d)$ e porre:

$$\begin{aligned} \alpha_{j,i} &= \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ i : \varphi(i, s, d) = j}} \pi_i(t, b_i-, s, d), \\ \alpha_{j,i} &= 0, \text{ if } j \notin r_j(i, s, d). \end{aligned}$$

Esempio. Si consideri una giunzione J con due linee entranti $\{1, 2\}$ e due linee uscenti $\{3, 4\}$ e si supponga che $r_j(1, s, d) = \{3, 4\}$ e $r_j(2, s, d) = \{3\}$. Dal momento che $\alpha_{4,2} = 0$, abbiamo $\alpha_{3,2} = 1$. I coefficienti $\alpha_{3,1}$ e $\alpha_{4,1}$ possono assumere i seguenti valori:

$$\begin{cases} \alpha_{3,1} = 0, \\ \alpha_{4,1} = 1, \end{cases} \text{ oppure } \begin{cases} \alpha_{3,1} = 1, \\ \alpha_{4,1} = 0. \end{cases}$$

Otteniamo un numero finiti di possibili matrici A :

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, A = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

Osservazione. Questo modello propone una strategia esclusiva. Infatti, tutti il flusso di pacchetti alla giunzione viene instradato dalla linea 1 alla linea 3 o alla linea 4.

Comunque, è più naturale assegnare una strategia flessibile, definendo un insieme di matrici ammissibili A nella seguente forma:

$$\mathbf{A} = \left\{ A : \exists \alpha_j^{i,s,d,j} \in [0,1], \sum_{j=n+1}^{n+m} \alpha_j^{i,s,d,j} = 1, \alpha_j^{i,s,d,j} = 0, \text{ if } j \notin r_j(i,s,d) : \right. \\ \left. \alpha_{j,i} = \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ j \in r_j(i,s,d)}} \pi_i(t, b_i-, s, d) \alpha_j^{i,s,d,j} \right\}$$

Infine, trattiamo ora il caso 2b).

In questo caso, la matrice A è unica ed è definita da:

$$\alpha_{j,i} = \sum_{s \in \mathbf{S}, d \in \mathbf{D}} \pi_i(t, b_i-, s, d) \alpha_j^{i,s,d,j}.$$

3.5 Riemann solvers alle giunzioni

In questa sezione, definiamo le soluzioni dei problemi di Riemann alle giunzioni, dal momento che questo è il punto di partenza per costruire le soluzioni del problema di Cauchy mediante l'algoritmo del wave – front tracking.

Descriviamo due diversi Riemann solver alle giunzioni, che rappresentano due diversi algoritmi di smistamento:

RA1 Assumiamo che:

- A. il traffico dalle linee di trasmissione entranti venga distribuito verso le linee di trasmissione uscenti in accordo a dei coefficienti fissati;
- B. nel rispetto della regola (A), il router sceglie di inoltrare i pacchetti in maniera tale da massimizzare il flusso (inteso come numero di pacchetti che devono essere processati) sulle linee entranti.

RA2 Assumiamo che il numero di pacchetti attraverso la giunzione venga massimizzato sia sulle linee entranti sia sulle linee uscenti.

Osservazione. Nel seguito, analizziamo il caso in cui le funzioni di distribuzione del traffico siano del tipo 2). Il caso 1) è stato considerato in [11] usando la seguente regola:

(RGP) Assumiamo che:

- A. il traffico dalle linee di trasmissione entranti viene distribuito sulle uscenti in accordo a dei coefficienti fissi;

B. nel rispetto della regola A. , il router sceglie di smistare i pacchetti per massimizzare

$$c_2 \sum_{i=1}^n f_i(\rho_i(\cdot, b_i-)) - c_1 [dist((f_1(\rho_1(\cdot, b_1-)), \dots, f_n(\rho_n(\cdot, b_n-))), r)]^2$$

con la condizione che

$$f_j(\rho_j(\cdot, a_j+)) = \sum_{i=1}^n \alpha_{j,i} f_i(\rho_i(\cdot, b_i-)), \quad \forall j = n+1, \dots, n+m,$$

dove c_1 e c_2 sono costanti strettamente positive, e $dist(\cdot, r)$ indica la distanza euclidea in R^n dalla retta r , che è data da

$$\begin{cases} \gamma_2 = p_1 \gamma_1, \\ \vdots \\ \gamma_n = p_{n-1} \gamma_{n-1}, \end{cases}$$

e $(p_1, \dots, p_{n-1})$ determina un livello di priorità per le linee entranti alle giunzioni. Questa procedura di massimizzazione considera priorità su linee entranti e assicura la continuità delle soluzioni rispetto ai coefficienti π .

3.5.1 Dinamiche ai nodi secondo l'RA1

Dobbiamo distinguere i casi 2a) e 2b).

Nel caso 2a), per risolvere il problema di Riemann alla giunzione, bisogna provare che la regione ammissibile è convessa. Proviamo dapprima il seguente lemma.

Lemma. *L'insieme $\mathbf{A}$ è convesso.*

Dimostrazione. Consideriamo una combinazione convessa $\lambda A_1 + (1-\lambda)A_2$ con $\lambda \in [0,1]$, $A_1, A_2 \in \mathbf{A}$. Abbiamo:

$$(\lambda A_1 + (1-\lambda)A_2)_{i,j} = \lambda \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ j \in r_j(i,s,d)}} \pi_i \alpha_{j,1}^{i,s,d,j} + (1-\lambda) \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ j \in r_j(i,s,d)}} \pi_i \alpha_{j,2}^{i,s,d,j} =$$

$$\sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ j \in r_j(i,s,d)}} \pi_i (\lambda \alpha_{j,1}^{i,s,d,j} + (1-\lambda) \alpha_{j,2}^{i,s,d,j}) = \sum_{\substack{s \in \mathbf{S}, d \in \mathbf{D}, \\ j \in r_j(i,s,d)}} \pi_i \hat{\alpha}_j^{i,s,d,j},$$

con $\hat{\alpha}_j^{i,s,d,j} \in [0,1]$. Inoltre,

$$\sum_{j=n+1}^{n+m} \hat{\alpha}_j^{i,s,d,j} = \sum_{j=n+1}^{n+m} (\lambda \alpha_{j,1}^{i,s,d,j} + (1-\lambda) \alpha_{j,2}^{i,s,d,j}) = \lambda \sum_{j=n+1}^{n+m} \alpha_{j,1}^{i,s,d,j} + (1-\lambda) \sum_{j=n+1}^{n+m} \alpha_{j,2}^{i,s,d,j} = 1,$$

allora $\lambda A_1 + (1-\lambda)A_2 \in \mathbf{A}$. ■

Si noti ora che la regione ammissibile è data da:

$$\Omega_{adm} = \{ \hat{\gamma} : \hat{\gamma} \in \Omega_1 \times \dots \times \Omega_n, \exists A \in \mathbf{A} \text{ t.c. } A\hat{\gamma} \in \Omega_{n+1} \times \dots \times \Omega_{n+m} \}.$$

Possiamo provare che tale regione è convessa almeno per il caso di giunzioni con due linee entranti e due linee uscenti. Più precisamente, abbiamo il seguente lemma:

Lemma. *Si consideri una giunzione J con $n=2$ linee entranti e $m=2$ linee uscenti e si assume che ci sia un'unica sorgente ed un'unica destinazione. Allora, l'insieme Ω_{adm} è convesso.*

Dimostrazione. Dobbiamo considerare i seguenti casi:

- i.) $r_j(1, s, d) = 3$ e $r_j(2, s, d) = 3$;
- ii.) $r_j(1, s, d) = 3$ e $r_j(2, s, d) = 4$;
- iii.) $r_j(1, s, d) = \{3, 4\}$ e $r_j(2, s, d) = 4$;
- iv.) $r_j(1, s, d) = \{3, 4\}$ e $r_j(2, s, d) = \{3, 4\}$.

Tutti gli altri casi possono essere ottenuto chiamando le linee in maniera diversa. I casi i) e ii) sono immediati, poichè $\hat{\gamma} \in \Omega_1 \times \Omega_2$ soddisfa $\hat{\gamma} \in \Omega_{adm}$ se e solo se $\hat{\gamma}_1 + \hat{\gamma}_2 \leq \gamma_3^{\max}$ (caso i)) o $\hat{\gamma}_1 \leq \gamma_3^{\max}$ e $\hat{\gamma}_2 \leq \gamma_4^{\max}$ (caso ii)).

Si consideri ora il caso iii). Allora, $A\hat{\gamma}$, $A \in \mathbf{A}$, è il segmento che unisce il punto $(\hat{\gamma}_1, \hat{\gamma}_2)$ e il punto $(0, \hat{\gamma}_1 + \hat{\gamma}_2)$. Si ha che $\hat{\gamma} \in \Omega_1 \times \Omega_2$ soddisfa $\hat{\gamma} \in \Omega_{adm}$ se e solo se $\hat{\gamma}_1 + \hat{\gamma}_2 \leq \gamma_3^{\max} + \gamma_4^{\max}$ e $\hat{\gamma}_2 \leq \gamma_3^{\max}$.

Infine, si assuma che valga il caso iv). Allora, $A\hat{\gamma}$, $A \in \mathbf{A}$, è il segmento che unisce il punto $(\hat{\gamma}_1 + \hat{\gamma}_2, 0)$ e il punto $(0, \hat{\gamma}_1 + \hat{\gamma}_2)$. Quindi $\hat{\gamma} \in \Omega_1 \times \Omega_2$ soddisfa $\hat{\gamma} \in \Omega_{adm}$ se e solo se $\hat{\gamma}_1 + \hat{\gamma}_2 \leq \gamma_3^{\max} + \gamma_4^{\max}$.

Se la regione Ω_{adm} è convessa, allora le regole (A) e (B) corrispondono al problema di programmazione lineare:

$$\max_{\hat{\gamma} \in \Omega_{adm}} (\hat{\gamma}_1 + \hat{\gamma}_2).$$

Tale problema ha chiaramente una soluzione, che può non essere unica. ■

Consideriamo il caso 2b). Introduciamo alcune notazioni:

Definizione. Sia $\tau : [0,1] \rightarrow [0,1]$ la mappa tale che:

- $f(\tau(\rho)) = f(\rho)$ per ogni $\rho \in [0,1]$;
- $\tau(\rho) \neq \rho$ per ogni $\rho \in [0,1] \setminus \{\sigma\}$.

Chiaramente, τ è ben definita e soddisfa:

$$\begin{aligned} 0 \leq \rho \leq \sigma &\Leftrightarrow \sigma \leq \tau(\rho) \leq 1, \\ \sigma \leq \rho \leq 1 &\Leftrightarrow 0 \leq \tau(\rho) \leq \sigma. \end{aligned}$$

Sia $\{e_1, \dots, e_n\}$ la base canonica di $\square^n$ e, per ogni sottoinsieme $V \subset \square^n$, sia $V^\perp$ il suo ortogonale. Si definisca, per ogni $i = 1, \dots, n$, $H_i = \{e_i\}^\perp$, ovvero l'iperpiano ortogonale a e_i e, per ogni $j = n+1, \dots, n+m$, sia $\alpha_j = \{\alpha_{j1}, \dots, \alpha_{jn}\} \in \square^n$ e si definisca $H_j = \{\alpha_j\}^\perp$. Sia $\mathbf{K}$ il set di indici $k = (k_1, \dots, k_l)$, $1 \leq l \leq n-1$, tale che $0 \leq k_1 < k_2 < \dots < k_l \leq n+m$ e per ogni $k \in \mathbf{K}$, sia $H_k = \bigcap_{h=1}^l H_{k_h}$.

Ponendo $\mathbf{1} = (1, \dots, 1) \in \square^n$, assumiamo che

(C) per ogni $k \in \mathbf{K}$, $\mathbf{1} \notin H_k^\perp$.

Nel caso 2b), sussiste il seguente risultato:

Teorema. Sia $(\mathbf{N}, \mathbf{I}, \mathbf{F}, \mathbf{J}, \mathbf{S}, \mathbf{D}, \mathbf{R})$ una rete ammissibile e J una giunzione con n linee entranti e m linee uscenti. Si assuma che il flusso $f : [0,1] \rightarrow R$ soddisfi (F) e che la matrice A soddisfi la condizione (C). Per ogni $\rho_{1,0}, \dots, \rho_{n+m,0} \in [0,1]$, e per ogni $\pi_1^{s,d}, \dots, \pi_{n+m}^{s,d} \in [0,1]$, esistono densità $\hat{\rho}_1, \dots, \hat{\rho}_{n+m}$ e un'unica soluzione ammissibile centrata, $\rho = (\rho_1, \dots, \rho_{n+m})$, alla giunzione J , tale che:

$$\begin{aligned} \rho_1(0, \cdot) &\equiv \rho_{1,0}, \dots, \rho_{n+m}(0, \cdot) \equiv \rho_{n+m,0}, \\ \pi^1(0, \cdot, s, d) &= \pi_1^{s,d}, \dots, \pi^{n+m}(0, \cdot, s, d) = \pi_{n+m}^{s,d}, (s \in \mathbf{S}, d \in \mathbf{D}). \end{aligned}$$

Abbiamo che:

$$\begin{aligned} \hat{\rho}_i &\in \begin{cases} \{\rho_{i,0}\} \cup [\tau(\rho_{i,0}), 1], & \text{se } 0 \leq \rho_{i,0} \leq \sigma, \\ [\sigma, 1], & \text{se } \sigma \leq \rho_{i,0} \leq 1, \end{cases} \quad i = 1, \dots, n, \\ \hat{\rho}_j &\in \begin{cases} [0, \sigma], & \text{se } 0 \leq \rho_{j,0} \leq \sigma, \\ \{\rho_{j,0}\} \cup [0, \tau(\rho_{j,0})], & \text{se } \sigma \leq \rho_{j,0} \leq 1, \end{cases} \quad j = n+1, \dots, n+m, \end{aligned}$$

e su ogni linea entrante I_i , $i = 1, \dots, n$, la soluzione consiste di una singola onda $(\rho_{i,0}, \hat{\rho}_i)$, mentre su ciascuna linea uscente I_j , $j = n+1, \dots, n+m$, la soluzione consiste della singola onda $(\hat{\rho}_j, \rho_{j,0})$.

Inoltre, $\hat{\pi}_i(t, \cdot, s, d) = \pi_i^{s,d}$ per ogni $t \geq 0, i \in \{1, \dots, n\}, s \in \mathbf{S}, d \in \mathbf{D}$ e

$$\hat{\pi}_j(t, a_j +, s, d) = \frac{\sum_{i=1}^n \alpha_j^{i,s,d,j} \pi_i^{s,d}(t, b_i -, s, d) f(\hat{\rho}_i)}{f(\hat{\rho}_j)},$$

per ogni $t \geq 0, j \in \{n+1, \dots, n+m\}, s \in \mathbf{S}, d \in \mathbf{D}$.

3.5.2 Dinamiche ai nodi secondo l'RA2

Per risolvere il problema di Riemann in accordo all'algoritmo (RA2), abbiamo bisogno di alcuni parametri addizionali chiamati parametri di “priorità” e di “distribuzione del traffico”. Per semplicità di esposizione, consideriamo prima una giunzione J in cui ci sono due linee di trasmissione con traffico entrante e due linee di trasmissione con traffico uscente. In tal caso, abbiamo solo un parametro di priorità $q \in]0, 1[$ ed un parametro di distribuzione del traffico $\alpha \in]0, 1[$. Denotiamo con $(\rho_{1,0}, \rho_{2,0}, \rho_{3,0}, \rho_{4,0})$ e $(\pi_{1,0}^{s,d}, \pi_{2,0}^{s,d}, \pi_{3,0}^{s,d}, \pi_{4,0}^{s,d})$ i dati iniziali.

Per massimizzare il numero di pacchetti che, attraverso una linea di trasmissione, entrano in una giunzione e da questa escono, definiamo

$$\Gamma = \min \{ \Gamma_{in}^{\max}, \Gamma_{out}^{\max} \},$$

dove $\Gamma_{in}^{\max} = \gamma_1^{\max} + \gamma_2^{\max}$ e $\Gamma_{out}^{\max} = \gamma_3^{\max} + \gamma_4^{\max}$. Dunque, vogliamo ottenere Γ come flusso attraverso la giunzione.

Si può facilmente notare che, per risolvere il problema di Riemann, basta determinare i flussi $\hat{\gamma}_i = f(\hat{\rho}_i)$, $i = 1, 2$. Infatti, per avere onde semplici con velocità appropriate, cioè negative sulle linee entranti e positive su quelle uscenti, assegniamo i vincoli (3.6), (3.7). Osserviamo che si può calcolare $\hat{\gamma}_i = f(\hat{\rho}_i)$, $i = 1, 2$ senza considerare il tipo di funzione di distribuzione del traffico.

Dobbiamo distinguere due casi:

I.) $\Gamma_{in}^{\max} = \Gamma$;

II.) $\Gamma_{in}^{\max} > \Gamma$.

Nel primo caso, poniamo $\hat{\gamma}_i = \gamma_i^{\max}$, $i = 1, 2$.

Analizziamo il secondo caso, in cui usiamo il parametro di priorità q . Non tutti i pacchetti possono

entrare nella giunzione, e sia C la quantità di pacchetti che può farlo. Dunque qC pacchetti provengono dalla prima linea entrante e $(1-q)C$ pacchetti dalla seconda.

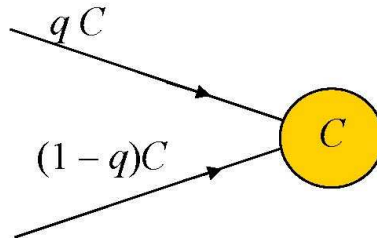


Figura 5.7: interpretazione del parametro di priorità.

Consideriamo lo spazio (γ_1, γ_2) e definiamo le seguenti rette:

$$r_q : \gamma_2 = \frac{1-q}{q} \gamma_1,$$

$$r_\Gamma : \gamma_1 + \gamma_2 = \Gamma.$$

Sia P il punto d'intersezione delle linee r_q e r_Γ . Ricordiamo che il flusso finale dovrebbe appartenere alla regione:

$$\Omega = \{(\gamma_1, \gamma_2) : 0 \leq \gamma_i \leq \gamma_i^{\max}, i=1,2\}.$$

Distinguiamo due casi:

- a) P appartiene a Ω ;
- b) P non appartiene a Ω .

Nel primo caso, scegliamo $(\hat{\gamma}_1, \hat{\gamma}_2) = P$, mentre, nel secondo caso, scegliamo $(\hat{\gamma}_1, \hat{\gamma}_2) = Q = \text{proj}_{\Omega \cap r_\Gamma}(P)$, dove proj è la proiezione canonica su un insieme convesso.

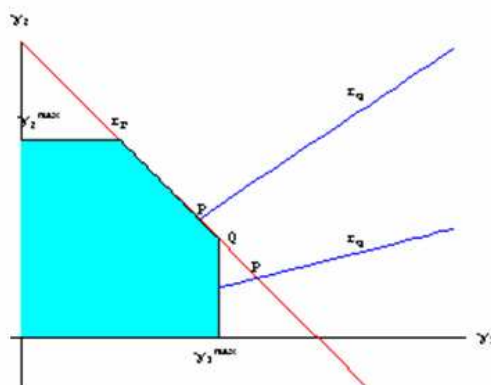


Figura 5.8: Riemann solver nei casi a) e b).

Il ragionamento si può ripetere anche nel caso di n linee entranti. In $\square^n$ la retta r_q è data da $r_q = tv_q$, $t \in \square$, con $v_q \in \Delta_{n-1}$ dove

$$\Delta_{n-1} = \left\{ (\gamma_1, \dots, \gamma_n) : \gamma_i \geq 0, i = 1, \dots, n, \sum_{i=1}^n \gamma_i = 1 \right\}$$

è il simpleso di dimensione $(n-1)$ e

$$H_\Gamma = \left\{ (\gamma_1, \dots, \gamma_n) : \sum_{i=1}^n \gamma_i = \Gamma \right\}$$

è un iperpiano con $\Gamma = \min\{\sum_{in} \gamma_i^{\max}, \sum_{out} \gamma_j^{\max}\}$. Poiché $v_q \in \Delta_{n-1}$, allora esiste un unico punto $P = r_q \cap H_\Gamma$. Se $P \in \Omega$, scegliamo $(\hat{\gamma}_1, \dots, \hat{\gamma}_n) = P$. Se $P \notin \Omega$, allora scegliamo $(\hat{\gamma}_1, \dots, \hat{\gamma}_n) = Q = \text{proj}_{\Omega \cap H_\Gamma}(P)$, la proiezione sull'insieme $\Omega \cap H_\Gamma$. Si osservi che la proiezione è unica poiché $\Omega \cap H_\Gamma$ è un sottoinsieme chiuso convesso di H_Γ .

Osservazione. Una possibile definizione alternativa nel caso $P \notin \Omega$ è scegliere $(\hat{\gamma}_1, \dots, \hat{\gamma}_n)$ come uno dei vertici di $\Omega \cap H_\Gamma$.

Come per l'algoritmo (RA1), $\hat{\pi}_i^{s,d} = \pi_{i,0}^{s,d}$, $i = 1, 2$.

Determiniamo, ora, $\hat{\gamma}_j$, $j = 3, 4$.

Come per le linee di trasmissione entranti, dobbiamo distinguere due casi:

$$\text{I.) } \Gamma_{out}^{\max} = \Gamma;$$

$$\text{II.) } \Gamma_{out}^{\max} > \Gamma.$$

Nel primo caso, $\hat{\gamma}_j = \gamma_j^{\max}$, $j = 3, 4$. Determiniamo $\hat{\gamma}_j$ nel secondo caso.

Riferiamoci ad α , il parametro di distribuzione del traffico. Poiché non tutti i pacchetti possono essere smistati sulle linee di trasmissione uscenti, sia C la quantità di pacchetti che può farlo. Dunque, αC pacchetti verranno instradati verso la linea uscente I_3 e $(1-\alpha)C$ verso la linea uscente I_4 . Si consideri lo spazio (γ_3, γ_4) e si definiscano le seguenti rette:

$$r_\alpha : \gamma_4 = \frac{1-\alpha}{\alpha} \gamma_3,$$

$$r_\Gamma : \gamma_3 + \gamma_4 = \Gamma.$$

La retta r_α può essere calcolata a partire dalla matrice A . Infatti, se assumiamo che la matrice di distribuzione del traffico A sia assegnata, allora calcoliamo $\hat{\gamma}_1, \dots, \hat{\gamma}_n$, e scegliamo $v_\alpha \in \Delta_{m-1}$ da

$$v_\alpha = \Delta_{m-1} \cap \{tA(\hat{\gamma}_1, \dots, \hat{\gamma}_n) : t \in \square\},$$

dove

$$\Delta_{m-1} = \left\{ (\gamma_{n+1}, \dots, \gamma_{m+n}) : \gamma_{n+i} \geq 0, i = 1, \dots, m, \sum_{i=1}^n \gamma_{n+i} = 1 \right\}$$

è il semplice $(m-1)$ dimensionale.

Se abbiamo un'unica matrice A ed un unico vettore v_α , allora i flussi sulle linee uscenti vengono calcolati come nel caso senza sorgente e destinazione ([9]).

Dobbiamo distinguere il caso 2a) e il caso 2b) per la funzione di distribuzione del traffico.

Caso 2a) Introduciamo l'insieme

$$\mathbf{G} = \{ A \hat{\gamma}_{inc}^T : A \in \mathbf{A} \}.$$

Lemma. L'insieme $\mathbf{G}$ è connesso.

Dimostrazione. L'insieme $\mathbf{G}$ è l'immagine di un insieme connesso attraverso una mappa continua.

Fissati $(\hat{\gamma}_1, \hat{\gamma}_2)$, la mappa è definita da:

$$(\tilde{\alpha}_j^{1,s,d,3}, \tilde{\alpha}_j^{2,s,d,3}) \in [0,1] \times [0,1] \rightarrow (\Sigma, \hat{\gamma}_1 + \hat{\gamma}_2 - \Sigma),$$

dove $\Sigma = \sum_{s,d} (\hat{\gamma}_1 \pi_1^{s,d} \tilde{\alpha}_j^{1,s,d,3} + \hat{\gamma}_2 \pi_2^{s,d} \tilde{\alpha}_j^{2,s,d,3})$. ■

Indichiamo con G_1 e G_2 i punti finali di tale insieme. Dal momento che, nel caso 2a), abbiamo un infinito numero di matrici A , ognuna delle quali determina una retta r_α , scegliamo la retta “più naturale” r_α , ovvero quella più vicina alla linea statistica determinata dalle misure sulla rete.

Si noti che i flussi finali devono appartenere alla regione:

$$\Omega = \{ (\gamma_3, \gamma_4) : 0 \leq \gamma_j \leq \gamma_j^{\max}, j = 3, 4 \}.$$

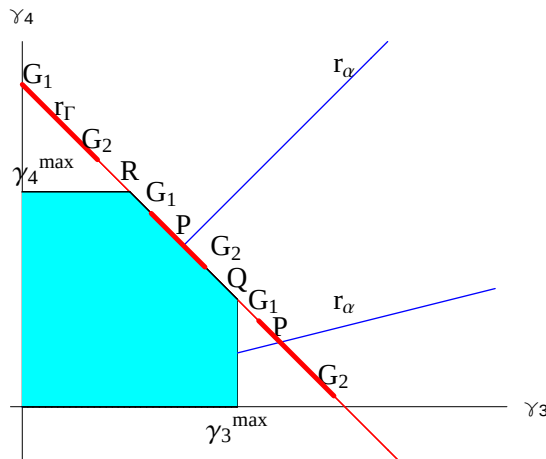


Figura 3.9: Funzione di distribuzione del traffico di tipo 2a).

Siano $P = r_\alpha \cap r_\Gamma$, $R = (\Gamma - \gamma_4^{\max}, \gamma_4^{\max})$, $Q = (\gamma_3^{\max}, \Gamma - \gamma_3^{\max})$. Distinguiamo tre casi:

- a) $\mathbf{G} \cap \Omega \cap r_\Gamma \neq \emptyset$,
- b) $\mathbf{G} \cap \Omega \cap r_\Gamma = \emptyset$ e $\gamma_3(G_1) < \gamma_3(R)$,
- c) $\mathbf{G} \cap \Omega \cap r_\Gamma = \emptyset$ e $\gamma_3(G_1) > \gamma_3^{\max}$.

Se l'insieme $\mathcal{G}$ ha una priorità sulla retta r_Γ deniamo $(\hat{\gamma}_3, \hat{\gamma}_4)$ nel seguente. Nel caso a), definiamo $(\hat{\gamma}_3, \hat{\gamma}_4) = \text{proj}_{\mathbf{G} \cap \Omega \cap r_\Gamma}(P)$, nel caso b) $(\hat{\gamma}_3, \hat{\gamma}_4) = R$, e infine nel caso c) $(\hat{\gamma}_3, \hat{\gamma}_4) = Q$.

Altrimenti, se r_Γ ha una priorità su $\mathbf{G}$, definiamo $(\hat{\gamma}_3, \hat{\gamma}_4) = \min_{\gamma \in \Omega} \mathbf{F}(\gamma, r_\alpha, \mathbf{G})$, dove $\mathbf{F}$ è un funzionale convesso che dipende da γ , r_α e dall'insieme $\mathbf{G}$ degli standard di instradamento. Una possibile scelta di $\mathbf{F}$ è $\mathbf{F} = d(\gamma, B)$, dove $B = w_1 r_\alpha + w_2 \int_{\mathbf{G}} r dr$, con w_1, w_2 numeri reali e d indica una distanza.

Il ragionamento può essere ripetuto anche nel caso di m linee uscenti.

Il vettore $\hat{\pi}_i^{s,d}$, $j = 3, 4$ viene valutato nello stesso modo visto per l'algoritmo (RA1).

Caso 2b) Distinguiamo due casi:

- a) P appartiene a Ω ;
- b) P non appartiene a Ω .

Nel primo caso, scegliamo $(\hat{\gamma}_3, \hat{\gamma}_4) = P$, mentre nel secondo caso scegliamo $(\hat{\gamma}_3, \hat{\gamma}_4) = Q$, dove $Q = \text{proj}_{\Omega_{adm}}(P)$. Possiamo estendere quanto detto al caso di m linee uscenti come per le linee entranti, definendo l'iperpiano $H_\Gamma = \{(\gamma_{n+1}, \dots, \gamma_{n+m}) : \sum_{j=n+1}^{n+m} \gamma_j = \Gamma\}$ e scegliendo un vettore $v_\alpha \in \Delta_{m-1}$.

Infine, definiamo $\hat{\pi}_i^{s,d}$, $j = 3, 4$, come nel caso 2a):

$$\hat{\pi}_j(t, a_j +, s, d) = \frac{\sum_{i=1}^n \alpha_j^{i,s,d,j} \pi_i^{s,d}(t, b_i -, s, d) f(\hat{\rho}_i)}{f(\hat{\rho}_j)},$$

per ogni $t \geq 0$, $j \in \{n+1, \dots, n+m\}$, $s \in \mathbf{S}$, $d \in \mathbf{D}$.

Osservazione. Osserviamo che nel caso dell'algoritmo (RA2) troviamo separatamente la soluzione per le linee entranti ed uscenti.

Osservazione. Se $\Gamma_{out}^{\max} < \Gamma_{in}^{\max}$ possiamo definire un diverso Riemann solver, considerando un ordine di priorità per l'instradamento dei pacchetti: $(s_{k_1}, d_{l_1}) = c_1, (s_{k_2}, d_{l_2}) = c_2, (s_{k_3}, d_{l_3}) = c_3, \dots$. I pacchetti vengono inviati fino a quando la quantità di pacchetti spedita è pari a:

$$\sum_{t=1}^{\bar{t}} \sum_{i=1}^n \pi_i^{c_t} \gamma_{i0}$$

dove $\bar{t}$ è il minimo tale che

$$\sum_{t=1}^{\bar{t}} \sum_{i=1}^n \pi_i^{c_t} \gamma_{i0} > \Gamma.$$

Definiamo $d = \Gamma - \sum_{t=1}^{\bar{t}} \sum_{i=1}^n \pi_i^{c_t} \gamma_{i0}$. Allora,

$$\begin{aligned} \hat{\gamma}_1 &= \sum_{t=1}^{\bar{t}-1} \pi_1^{c_t} \gamma_{10} + d/2, \\ \hat{\gamma}_2 &= \sum_{t=1}^{\bar{t}-1} \pi_2^{c_t} \gamma_{20} + d/2. \end{aligned}$$

Una volta definite le soluzioni al problema di Riemann, può essere usato l'algoritmo del wave – front tracking per costruire una sequenza di soluzioni approssimate.

4 Bibliografia

- [1] D. Alderson, H. Chang, M. Roughan, S. Uhlig and W. Willinger, The many facets of internet topology and traffic, *Networks and Heterogeneous Media (NHM)*, Vol. 1, Issue 4, (2007), pp. 569 – 600 .
- [2] D. Armbruster, P. Degond and C. Ringhofer, A Model for the Dynamics of large Queuing Networks and Supply Chains, *SIAM J. Applied Mathematics, SIAP*, Vol. 66, Issue 3, (2006), pp. 896 – 920.
- [3] F. Baccelli, D. Hong and Z. Liu, Fixed points methods for the simulation of the sharing of a local loop by large number of interacting TCP connections, *Tech. Rep. RR-4154, INRIA*, April 2001.
- [4] C. Bardos and A. Y. Le Roux and J. C. Nedelec, First Order Quasilinear Equations with Boundary Conditions, *Commun. Partial Differential Equations*, 4 (1979), pp. 1017 – 1034.
- [5] A. Bressan, *Hyperbolic Systems of Conservation Laws - The One-dimensional Cauchy Problem*, Oxford Univ. Press, 2000.
- [6] G. Coclite, M. Garavello, B. Piccoli, Traffic Flow on Road Networks, *SIAM Journal on Mathematical Analysis* 36 (2005), 1862 - 1886.
- [7] C. Dafermos, *Hyperbolic Conservation Laws in Continuum Physics*, Springer-Verlag, 1999.
- [8] C. Daganzo, *Fundamentals of Transportation and Traffic Operations*, Pergamon-Elsevier, Oxford, 1997.
- [9] C. D'Apice, R. Manzo, B. Piccoli, Packets flow on telecommunication networks, *SIAM Journal of Mathematical Analysis*, Vol. 38, n. 3, (2006), pp. 717--740.
- [10] M. Garavello and B. Piccoli, *Traffic flow on networks*, AIMS book series on Applied Mathematics, 2006.
- [11] M. Garavello, B. Piccoli, Source-Destination Flow on a Road Network, *Communications Mathematical Sciences* 3 (2005), pp. 261--283.
- [12] M. Herty and A. Klar, Modeling, simulation, and optimization of traffic flow networks, *SIAM J. Sci. Comput.*, 25 (2003), pp. 1066--1087.
- [13] H. Holden and N.H. Risebro, A mathematical model of traffic flow on a network of unidirectional roads, *SIAM J. Math. Anal.*, 26 (1995), pp. 999--1017.
- [14] F. Kelly, A.K. Maulloo and D.K.H. Tan, Rate control in communication networks: shadow prices, proportional fairness and stability, *J. of the Operational Research Society*, 49 (1998), pp. 237 – 252.

- [15] M. J. Lighthill and G. B. Whitham, On kinetic waves. II. Theory of Traffic Flows on Long Crowded Roads, Proc. Roy. Soc. London Ser. A, 229 (1955), pp. 317--345.
- [16] G.F. Newell, Traffic Flow on Transportation Networks, MIT Press, Cambridge (MA,USA), 1980.
- [17] P. I. Richards, Shock Waves on the Highway, Oper. Res., 4, (1956), pp. 42--51.
- [18] D. Serre, Systems of conservation laws I and II, Cambridge University Press, 1999.
- [19] A.S. Tanenbaum, Computer Networks, Prentice Hall, Upper Saddle River, 2003.
- [20] W. Willinger and V. Paxson, Where Mathematics meets the Internet, Notices of the AMS, 45 (1998), pp. 961--970.