

NAG Library Routine Document

E04MGF/E04MGA

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

To supply optional parameters to E04MFF/E04MFA from an external file. More precisely, E04MGF must be used to supply optional parameters to E04MFF and E04MGA must be used to supply optional parameters to E04MFA.

E04MGA is a version of E04MGF that has additional parameters in order to make it safe for use in multithreaded applications (see Section 5). The initialization routine E04WBF **must** have been called before calling E04MGA.

2 Specification

2.1 Specification for E04MGF

```
SUBROUTINE E04MGF(IOPTNS, INFORM)
  INTEGER          IOPTNS, INFORM
```

2.2 Specification for E04MGA

```
SUBROUTINE E04MGA(IOPTNS, LWSAV, IWSAV, RWSAV, INFORM)
  INTEGER          IOPTNS, IWSAV(610), INFORM
  double precision RWSAV(475)
  LOGICAL          LWSAV(120)
```

3 Description

E04MGF/E04MGA may be used to supply values for optional parameters to E04MFF/E04MFA. E04MGF/E04MGA reads an external file and each line of the file defines a single optional parameter. It is only necessary to supply values for those parameters whose values are to be different from their default values.

Each optional parameter is defined by a single character string, of up to 72 characters, consisting of one or more items. The items associated with a given option must be separated by spaces, or equals signs [=]. Alphabetic characters may be upper or lower case. The string

```
Print Level = 1
```

is an example of a string used to set an optional parameter. For each option the string contains one or more of the following items:

- a mandatory keyword;
- a phrase that qualifies the keyword;
- a number that specifies an INTEGER or ***double precision*** value. Such numbers may be up to 16 contiguous characters in Fortran's I, F, E or D formats, terminated by a space if this is not the last item on the line.

Blank strings and comments are ignored. A comment begins with an asterisk (*) and all subsequent characters in the string are regarded as part of the comment.

The file containing the options must start with `Begin` and must finish with `End`. An example of a valid options file is:

```
Begin * Example options file
  Print level = 5
End
```

For E04MGF each line of the file is normally printed as it is read, on the current advisory message unit (see X04ABF), but printing may be suppressed using the keyword `Nolist`. To suppress printing of `Begin`, `Nolist` must be the first option supplied as in the file:

```
Begin
  Nolist
  Print level = 5
End
```

Printing will automatically be turned on again after a call to E04MFF or E04MGF and may be turned on again at any time using the keyword `List`.

For E04MGA printing is turned off by default, but may be turned on at any time using the keyword `List`.

Optional parameter settings are preserved following a call to E04MFF/E04MFA and so the keyword `Defaults` is provided to allow you to reset all the optional parameters to their default values before a subsequent call to E04MFF/E04MFA.

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 11 in E04MFF/E04MFA.

4 References

None.

5 Parameters

1: IOPTNS – INTEGER *Input*

On entry: the unit number of the options file to be read.

Constraint: $0 \leq \text{IOPTNS} \leq 99$.

2: INFORM – INTEGER *Output*

Note: for E04MGA, *INFORM* does not occur in this position in the parameter list. See the additional parameters described below.

On exit: contains zero if the options file has been successfully read and a value > 0 otherwise (see Section 6).

Note: the following are additional parameters for specific use with E04MGA. Users of E04MGF therefore need not read the remainder of this description.

2: LWSAV(120) – LOGICAL array *Communication Array*

3: IWSAV(610) – INTEGER array *Communication Array*

4: RWSAV(475) – *double precision* array *Communication Array*

The arrays LWSAV, IWSAV and RWSAV **must not** be altered between calls to any of the routines E04MGA, E04MFA, E04MHA or E04WBF.

5: INFORM – INTEGER *Output*

Note: see the parameter description for INFORM above.

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFORM = 1

IOPTNS is not in the range [0,99].

INFORM = 2

Begin was found, but end-of-file was found before End was found.

INFORM = 3

end-of-file was found before Begin was found.

INFORM = 4

Not used.

INFORM = 5

One or more lines of the options file is invalid. Check that all keywords are neither ambiguous nor misspelt.

7 Accuracy

Not applicable.

8 Further Comments

E04MHF/E04MHA may also be used to supply optional parameters to E04MFF/E04MFA.

9 Example

This example solves the same problem as the example for E04MFF/E04MFA, but in addition illustrates the use of E04MGF/E04MGA and E04MHF/E04MHA to set optional parameters for E04MFF/E04MFA.

In this example the options file read by E04MGF/E04MGA is appended to the data file for the program (see Section 9.2). It would usually be more convenient in practice to keep the data file and the options file separate.

9.1 Program Text

Note: the following program illustrates the use of E04MGF. An equivalent program illustrating the use of E04MGA is available with the supplied Library and is also available from the NAG web site.

```
*      E04MGF Example Program Text
*      Mark 18 Revised. NAG Copyright 1997.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, NCMAX
      PARAMETER        (NMAX=10,NCMAX=10)
      INTEGER          LDA
      PARAMETER        (LDA=NCMAX)
      INTEGER          LIWORK, LWORK
      PARAMETER        (LIWORK=1000,LWORK=10000)
*      .. Local Scalars ..
      DOUBLE PRECISION OBJ
      INTEGER          I, IFAIL, INFORM, ITER, J, N, NCLIN, OUTCHN
      LOGICAL          LMOK
*      .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), AX(NCMAX), BL(NMAX+NCMAX),
+                     BU(NMAX+NCMAX), CLAMDA(NMAX+NCMAX), CVEC(NMAX),
```

```

+          WORK(LWORK), X(NMAX)
  INTEGER      ISTATE(NMAX+NCMAX), IWORK(LIWORK)
*
* .. External Subroutines ..
  EXTERNAL     E04MFF, E04MGF, E04MHF, X04ABF
*
* .. External Functions ..
  LOGICAL      A00ACF
  EXTERNAL     A00ACF
*
* .. Executable Statements ..
  WRITE (NOUT,*) 'E04MGF Example Program Results'
  OUTCHN = NOUT
  LMOK = A00ACF()
  IF (LMOK) THEN
*
*   Skip heading in data file
  READ (NIN,*)
  READ (NIN,*) N, NCLIN
  IF (N.LE.NMAX .AND. NCLIN.LE.NCMAX) THEN
*
*   Read CVEC, A, BL, BU and X from data file
*
  READ (NIN,*) (CVEC(I),I=1,N)
  READ (NIN,*) ((A(I,J),J=1,N),I=1,NCLIN)
  READ (NIN,*) (BL(I),I=1,N+NCLIN)
  READ (NIN,*) (BU(I),I=1,N+NCLIN)
  READ (NIN,*) (X(I),I=1,N)
*
*   Set three options using E04MHF
*
  CALL E04MHF(' Print Level = 1 ')
*
  CALL E04MHF(' Check Frequency = 10 ')
*
  CALL E04MHF(' Infinite Bound Size = 1.0D+25 ')
*
*   Set the unit number for advisory messages to OUTCHN
*
  CALL X04ABF(1,OUTCHN)
*
*   Read the options file for the remaining options
*
  CALL E04MGF(NIN,INFORM)
*
  IF (INFORM.NE.0) THEN
+      WRITE (NOUT,99999) 'E04MGF terminated with INFORM = ',
+      INFORM
  ELSE
*
*   Solve the problem
*
  IFAIL = 1
*
+      CALL E04MFF(N,NCLIN,A,LDA,BL,BU,CVEC,ISTATE,X,ITER,OBJ,
+      AX,CLAMDA,IWORK,LIWORK,WORK,LWORK,IFAIL)
  END IF
*
  ELSE
    WRITE (NOUT,*)
    WRITE (NOUT,*) 'At least one of N or NCLIN is too large'
  END IF
  ELSE
    WRITE (NOUT,*)
    WRITE (NOUT,*) ' ** A valid licence key was not found'
  END IF
*
99999 FORMAT (1X,A,I3)
END

```

9.2 Program Data

E04MGF Example Program Data

```

7 7                                     :Values of N and NCLIN
-0.02 -0.20 -0.20 -0.20 -0.20 0.04 0.04 :End of CVEC
1.00 1.00 1.00 1.00 1.00 1.00 1.00
0.15 0.04 0.02 0.04 0.02 0.01 0.03
0.03 0.05 0.08 0.02 0.06 0.01 0.00
0.02 0.04 0.01 0.02 0.02 0.00 0.00
0.02 0.03 0.00 0.00 0.01 0.00 0.00
0.70 0.75 0.80 0.75 0.80 0.97 0.00
0.02 0.06 0.08 0.12 0.02 0.01 0.97 :End of matrix A
-0.01 -0.10 -0.01 -0.04 -0.10 -0.01 -0.01
-0.13 -1.0D+25 -1.0D+25 -1.0D+25 -1.0D+25 -9.92D-02 -3.0D-03 :End of BL
0.01 0.15 0.03 0.02 0.05 1.0D+25 1.0D+25
-0.13 -4.9D-03 -6.4D-03 -3.7D-03 -1.2D-03 1.0D+25 2.0D-03 :End of BU
-0.01 -0.03 0.00 -0.01 -0.10 0.02 0.01 :End of X
Begin Example options file for E04MGF
  Crash Tolerance = 0.05 * (Default = 0.01)
  Iteration Limit = 25 * (Default = 70)
End
```

9.3 Program Results

E04MGF Example Program Results

Calls to E04MHF

```

Print Level = 1
Check Frequency = 10
Infinite Bound Size = 1.0D+25
```

OPTIONS file

```

Begin Example options file for E04MGF
  Crash Tolerance = 0.05 * (Default = 0.01)
  Iteration Limit = 25 * (Default = 70)
End
```

*** E04MFF

Parameters

```

Problem type..... LP

Linear constraints..... 7      Feasibility tolerance.. 1.05E-08
Variables..... 7      Optimality tolerance... 1.05E-08

Infinite bound size.... 1.00E+25      COLD start.....
Infinite step size.... 1.00E+25      EPS (machine precision) 1.11E-16

Check frequency..... 10      Expand frequency..... 5
Minimum sum of infeas.. NO      Crash tolerance..... 5.00E-02

Print level..... 1      Iteration limit..... 25
Monitoring file..... -1

Workspace provided is IWORK( 1000), WORK( 10000).
To solve problem we need IWORK( 17), WORK( 182).
```

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Slack
V 1	LL	-1.000000E-02	-1.000000E-02	1.000000E-02	0.3301	.
V 2	LL	-0.100000	-0.100000	0.150000	1.4384E-02	.

V	3	UL	3.000000E-02	-1.000000E-02	3.000000E-02	-9.0997E-02	.
V	4	UL	2.000000E-02	-4.000000E-02	2.000000E-02	-7.6612E-02	.
V	5	FR	-6.748534E-02	-0.100000	5.000000E-02	.	3.2515E-02
V	6	FR	-2.280130E-03	-1.000000E-02	None	.	7.7199E-03
V	7	FR	-2.345277E-04	-1.000000E-02	None	.	9.7655E-03

L	Con	State	Value	Lower Bound	Upper Bound	Lagr Mult	Slack
L	1	EQ	-0.130000	-0.130000	-0.130000	-1.431	2.7756E-17
L	2	FR	-5.479544E-03	None	-4.900000E-03	.	5.7954E-04
L	3	FR	-6.571922E-03	None	-6.400000E-03	.	1.7192E-04
L	4	FR	-4.849707E-03	None	-3.700000E-03	.	1.1497E-03
L	5	FR	-3.874853E-03	None	-1.200000E-03	.	2.6749E-03
L	6	LL	-9.920000E-02	-9.920000E-02	None	1.501	.
L	7	LL	-3.000000E-03	-3.000000E-03	2.000000E-03	1.517	-8.6736E-19

Exit E04MFF - Optimal LP solution.

Final LP objective value = 0.2359648E-01

Exit from LP problem after 5 iterations.