F11DEF – NAG Fortran Library Routine Document

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

F11DEF solves a real sparse nonsymmetric system of linear equations, represented in coordinate storage format, using a restarted generalized minimal residual (RGMRES), conjugate gradient squared (CGS), stabilized bi-conjugate gradient (Bi-CGSTAB), or transpose-free quasi-minimal residual (TFQMR) method, without preconditioning, with Jacobi, or with SSOR preconditioning.

2 Specification

```
 SUBROUTINE F11DEF(METHOD, PRECON, N, NNZ, A, IROW, ICOL, OMEGA, B,
1                  M, TOL, MAXITN, X, RNORM, ITN, WORK, LWORK,
2                  IWORK, IFAIL)
 INTEGER           N, NNZ, IROW(NNZ), ICOL(NNZ), M, MAXITN, ITN,
1                  IWORK(2*N+1), LWORK, IFAIL
 real              A(NNZ), OMEGA, B(N), TOL, X(N), RNORM,
1                  WORK(LWORK)
 CHARACTER*(*)     METHOD
 CHARACTER*1       PRECON
```

3 Description

This routine solves a real sparse nonsymmetric system of linear equations:

$$Ax = b,$$

using an RGMRES [1], CGS [3], Bi-CGSTAB(ℓ) [4], [2], or TFQMR [6], [7] method.

The routine allows the following choices for the preconditioner:

no preconditioning;

Jacobi preconditioning [5];

symmetric successive-over-relaxation (SSOR) preconditioning [5].

For incomplete LU (ILU) preconditioning see F11DCF.

The matrix A is represented in coordinate storage (CS) format (see Section 2.1.1 of the Chapter Introduction) in the arrays A, IROW and ICOL. The array A holds the non-zero entries in the matrix, while IROW and ICOL hold the corresponding row and column indices.

F11DEF is a black-box routine which calls F11BDF, F11BEF and F11BFF. If you wish to use an alternative storage scheme, preconditioner, or termination criterion, or require additional diagnostic information, you should call these underlying routines directly.

4 References

[1] Saad Y and Schultz M (1986) GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems *SIAM J. Sci. Statist. Comput.* **7** 856–869

[2] Sleijpen G L G and Fokkema D R (1993) BiCGSTAB(ℓ) for linear equations involving matrices with complex spectrum *ETNA* **1** 11–32

[3] Sonneveld P (1989) CGS, a fast Lanczos-type solver for nonsymmetric linear systems *SIAM J. Sci. Statist. Comput.* **10** 36–52

[4] van der Vorst H (1989) Bi–CGSTAB, A fast and smoothly converging variant of Bi–CG for the solution of nonsymmetric linear systems *SIAM J. Sci. Statist. Comput.* **13** 631–644

[5] Young D (1971) *Iterative Solution of Large Linear Systems* Academic Press, New York

[6] Freund R W and Nachtigal N (1991) QMR: a Quasi-Minimal Residual Method for Non-Hermitian Linear Systems *Numer. Math.* **60** 315–339

[7] Freund R W (1993) A Transpose-Free Quasi-Mimimal Residual Algorithm for Non-Hermitian Linear Systems *SIAM J. Sci. Comput.* **14** 470–482

5 Parameters

1: METHOD — CHARACTER*(*) *Input*

On entry: the iterative method to be used. The possible choices are:

'RGMRES'	restarted generalized minimum residual method;
'CGS'	conjugate gradient squared method;
'BICGSTAB'	bi-conjugate gradient stabilized (ℓ) method;
'TFQMR'	transpose-free quasi-minimal residual method.

Constraint: METHOD = 'RGMRES', 'CGS', 'BICGSTAB', or 'TFQMR'.

2: PRECON — CHARACTER*1 *Input*

On entry: specifies the type of preconditioning to be used. The possible choices are:

'N' no preconditioning;
'J' Jacobi;
'S' symmetric successive-over-relaxation.

Constraint: PRECON = 'N', 'J', or 'S'.

3: N — INTEGER *Input*

On entry: n, the order of the matrix A.

Constraint: $\mathrm{N} \geq 1$.

4: NNZ — INTEGER *Input*

On entry: the number of non-zero elements in the matrix A.

Constraint: $1 \leq \mathrm{NNZ} \leq \mathrm{N}^2$.

5: A(NNZ) — ***real*** array *Input*

On entry: the non-zero elements of the matrix A, ordered by increasing row index, and by increasing column index within each row. Multiple entries for the same row and column indices are not permitted. The routine F11ZAF may be used to order the elements in this way.

6: IROW(NNZ) — INTEGER array *Input*

7: ICOL(NNZ) — INTEGER array *Input*

On entry: the row and column indices of the non-zero elements supplied in A.

Constraints: IROW and ICOL must satisfy the following constraints (which may be imposed by a call to F11ZAF):

$1 \leq \mathrm{IROW}(i) \leq \mathrm{N}$ and $1 \leq \mathrm{ICOL}(i) \leq \mathrm{N}$, for $i = 1, 2, \ldots,$NNZ.
$\mathrm{IROW}(i-1) < \mathrm{IROW}(i)$, or
$\mathrm{IROW}(i-1) = \mathrm{IROW}(i)$ and $\mathrm{ICOL}(i-1) < \mathrm{ICOL}(i)$, for $i = 2, 3, \ldots,$NNZ.

8: OMEGA — ***real*** *Input*

On entry: if PRECON = 'S', OMEGA is the relaxation parameter ω to be used in the SSOR method. Otherwise OMEGA need not be initialized and is not referenced.

Constraint: $0.0 <$ OMEGA < 2.0.

9: B(N) — ***real*** array *Input*

On entry: the right-hand side vector b.

10: M — INTEGER *Input*

On entry: if METHOD = 'RGMRES', M is the dimension of the restart subspace;
if METHOD = 'BICGSTAB', M is the order ℓ of the polynomial Bi-CGSTAB method;
otherwise, M is not referenced.

Constraints:

if METHOD = 'RGMRES', $0 <$ M $\leq \min$(N,50);
if METHOD = 'BICGSTAB', $0 <$ M $\leq \min$(N,10).

11: TOL — ***real*** *Input*

On entry: the required tolerance. Let x_k denote the approximate solution at iteration k, and r_k the corresponding residual. The algorithm is considered to have converged at iteration k if:

$$\|r_k\|_\infty \leq \tau \times (\|b\|_\infty + \|A\|_\infty \|x_k\|_\infty).$$

If TOL ≤ 0.0, $\tau = \max(\sqrt{\epsilon}, \sqrt{n}\,\epsilon)$ is used, where ϵ is the ***machine precision***. Otherwise $\tau = \max(\text{TOL}, 10\epsilon, \sqrt{n}\,\epsilon)$ is used.

Constraint: TOL < 1.0.

12: MAXITN — INTEGER *Input*

On entry: the maximum number of iterations allowed.

Constraint: MAXITN ≥ 1.

13: X(N) — ***real*** array *Input/Output*

On entry: an initial approximation to the solution vector x.

On exit: an improved approximation to the solution vector x.

14: RNORM — ***real*** *Output*

On exit: the final value of the residual norm $\|r_k\|_\infty$, where k is the output value of ITN.

15: ITN — INTEGER *Output*

On exit: the number of iterations carried out.

16: WORK(LWORK) — ***real*** array *Workspace*

17: LWORK — INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F11DEF is called.

Constraint:

if METHOD = 'RGMRES' then LWORK $\geq 4 \times$ N + M $\times$ (M+N+5) + ν + 101;
if METHOD = 'CGS' then LWORK $\geq 8 \times$ N + ν + 100;
if METHOD = 'BICGSTAB' then LWORK $\geq 2 \times$ N $\times$ (M+3) + M $\times$ (M+2) + ν + 100;
if METHOD = 'TFQMR' then LWORK $\geq 11 \times$ N + ν + 100;

where ν = N for PRECON = 'J' or 'S', and 0 otherwise.

18: IWORK(2∗N+1) — INTEGER array *Workspace*

19: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Errors and Warnings

If on entry IFAIL = 0 or −1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors detected by the routine:

IFAIL = 1

On entry, METHOD ≠ 'RGMRES', 'CGS', 'BICGSTAB', or 'TFQMR',
or PRECON ≠ 'N', 'J' or 'S',
or N < 1,
or NNZ < 1,
or NNZ > N^2,
or PRECON = 'S' and OMEGA lies outside the interval (0.0,2.0),
or M < 1,
or M > min(N,50), with METHOD = 'RGMRES',
or M > min(N,10), with METHOD = 'BICGSTAB',
or TOL ≥ 1.0,
or MAXITN < 1,
or LWORK too small.

IFAIL = 2

On entry, the arrays IROW and ICOL fail to satisfy the following constraints:

$1 \leq \mathrm{IROW}(i) \leq \mathrm{N}$ and $1 \leq \mathrm{ICOL}(i) \leq \mathrm{N}$, for $i = 1, 2, \ldots,$NNZ.
$\mathrm{IROW}(i-1) < \mathrm{IROW}(i)$, or
$\mathrm{IROW}(i-1) = \mathrm{IROW}(i)$ and $\mathrm{ICOL}(i-1) < \mathrm{ICOL}(i)$, for $i = 2, 3, \ldots,$NNZ.

Therefore a non-zero element has been supplied which does not lie within the matrix A, is out of order, or has duplicate row and column indices. Call F11ZAF to reorder and sum or remove duplicates.

IFAIL = 3

On entry, the matrix A has a zero diagonal element. Jacobi and SSOR preconditioners are not appropriate for this problem.

IFAIL = 4

The required accuracy could not be obtained. However, a reasonable accuracy may have been obtained, and further iterations could not improve the result. You should check the output value of RNORM for acceptability. This error code usually implies that your problem has been fully and satisfactorily solved to within or close to the accuracy available on your system. Further iterations are unlikely to improve on this situation.

IFAIL = 5

Required accuracy not obtained in MAXITN iterations.

IFAIL = 6

A serious error has occurred in an internal call to F11BDF, F11BEF or F11BFF. Check all subroutine calls and array sizes. Seek expert help.

7 Accuracy

On successful termination, the final residual $r_k = b - Ax_k$, where $k =$ ITN, satisfies the termination criterion

$$\|r_k\|_\infty \leq \tau \times (\|b\|_\infty + \|A\|_\infty \|x_k\|_\infty).$$

The value of the final residual norm is returned in RNORM.

8 Further Comments

The time taken by F11DEF for each iteration is roughly proportional to NNZ.

The number of iterations required to achieve a prescribed accuracy cannot be easily determined a priori, as it can depend dramatically on the conditioning and spectrum of the preconditioned coefficient matrix $\bar{A} = M^{-1}A$.

9 Example

This example program solves a sparse nonsymmetric system of equations using the RGMRES method, with SSOR preconditioning.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F11DEF Example Program Text
*     Mark 19 Revised. NAG Copyright 1999.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX, LA, LWORK
      PARAMETER        (NMAX=1000,LA=10000,LWORK=10000)
*     .. Local Scalars ..
      real             OMEGA, RNORM, TOL
      INTEGER          I, IFAIL, ITN, L, LWREQ, M, MAXITN, N, NNZ
      CHARACTER        PRECON
      CHARACTER*8      METHOD
*     .. Local Arrays ..
      real             A(LA), B(NMAX), WORK(LWORK), X(NMAX)
      INTEGER          ICOL(LA), IROW(LA), IWORK(2*NMAX+1)
*     .. External Subroutines ..
      EXTERNAL         F11DEF
*     .. Intrinsic Functions ..
      INTRINSIC        MAX
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F11DEF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
*
*     Read algorithmic parameters
*
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
         READ (NIN,*) NNZ
         READ (NIN,*) METHOD, PRECON
         READ (NIN,*) OMEGA
```

```
         READ (NIN,*) M, TOL, MAXITN
*
*        Check size of workspace
*
         L = N
         IF (PRECON.EQ.'N' .OR. PRECON.EQ.'n') L = 0
         LWREQ = MAX(4*N+M*(M+N+5)+L+101,8*N+L+100,2*N*(M+3)+M*(M+2)
     +           +L+100,11*N+L+100)
         IF (LWORK.LT.LWREQ) THEN
            WRITE (NOUT,*) 'LWORK must be at least', LWREQ
            STOP
         END IF
*
*        Read the matrix A
*
         DO 20 I = 1, NNZ
            READ (NIN,*) A(I), IROW(I), ICOL(I)
   20    CONTINUE
*
*        Read right-hand side vector b and initial approximate solution x
*
         READ (NIN,*) (B(I),I=1,N)
         READ (NIN,*) (X(I),I=1,N)
*
*        Solve Ax = b using F11DEF
*
         IFAIL = 0
         CALL F11DEF(METHOD,PRECON,N,NNZ,A,IROW,ICOL,OMEGA,B,M,TOL,
     +               MAXITN,X,RNORM,ITN,WORK,LWORK,IWORK,IFAIL)
*
         WRITE (NOUT,'(A,I10,A)') ' Converged in', ITN, ' iterations'
         WRITE (NOUT,'(A,1P,D16.3)') ' Final residual norm =', RNORM
         WRITE (NOUT,*)
*
*        Output x
*
         WRITE (NOUT,*) '           X'
         DO 40 I = 1, N
            WRITE (NOUT,'(1X,1P,D16.4)') X(I)
   40    CONTINUE
*
      END IF
      STOP
      END
```

9.2 Program Data

```
F11DEF Example Program Data
  5                  N
 16                  NNZ
  'RGMRES' 'S'       METHOD, PRECON
  1.05                OMEGA
  1 1.E-10 1000      M, TOL, MAXITN
  2.   1    1
  1.   1    2
 -1.   1    4
 -3.   2    2
 -2.   2    3
```

```
 1.   2    5
 1.   3    1
 5.   3    3
 3.   3    4
 1.   3    5
-2.   4    1
-3.   4    4
-1.   4    5
 4.   5    2
-2.   5    3
-6.   5    5          A(I), IROW(I), ICOL(I), I=1,...,NNZ
 0.  -7.  33.
-19. -28.             B(I), I=1,...,N
 0.   0.   0.
 0.   0.             X(I), I=1,...,N
```

9.3 Program Results

```
F11DEF Example Program Results

Converged in        13 iterations
Final residual norm =       5.087E-09

          X
     1.0000E+00
     2.0000E+00
     3.0000E+00
     4.0000E+00
     5.0000E+00
```