F11BSF – 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

F11BSF is an iterative solver for a complex general (non-Hermitian) system of simultaneous linear equations; F11BSF is the second in a suite of three routines, where the first routine, F11BRF, must be called prior to F11BSF to set-up the suite, and the third routine in the suite, F11BTF, can be used to return additional information about the computation.

These three routines are suitable for the solution of large sparse general (non-Hermitian) systems of equations.

2 Specification

```
SUBROUTINE F11BSF(IREVCM, U, V, WGT, WORK, LWORK, IFAIL)
INTEGER           IREVCM, LWORK, IFAIL
real              WGT(*)
complex           U(*), V(*), WORK(LWORK)
```

3 Description

F11BSF solves the general (non-Hermitian) system of linear simultaneous equations $Ax = b$ using one of four available methods: RGMRES, the preconditioned restarted generalized minimum residual method (Saad and Schultz [1]); CGS, the preconditioned conjugate gradient squared method (Sonneveld [2]); Bi-CGSTAB (ℓ), the bi-conjugate gradient stabilized method of order ℓ (Van der Vorst [3], Sleijpen and Fokkema [4]); or TFQMR, the quasi-minimal transpose-free quasi-minimum residual method (Freund and Nachtigal [6], Freund [7]).

For a general description of the methods employed you are referred to Section 3 of the document for F11BRF.

F11BSF can solve the system after the first routine in the suite, F11BRF, has been called to initialize the computation and specify the method of solution. The third routine in the suite, F11BTF, can be used to return additional information generated by the computation during monitoring steps and after F11BSF has completed its tasks.

F11BSF uses **reverse communication**, i.e., it returns repeatedly to the calling program with the parameter IREVCM (see Section 5) set to specified values which require the calling program to carry out one of the following tasks:

compute the matrix-vector product $v = Au$ or $v = A^H u$ (the four methods require the matrix transpose-vector product only if $\|A\|_1$ or $\|A\|_\infty$ is estimated internally by Higham's method (Higham [5]));

solve the preconditioning equation $Mv = u$;

notify the completion of the computation;

allow the calling program to monitor the solution.

Through the parameter IREVCM the calling program can cause immediate or tidy termination of the execution. On final exit, the last iterates of the solution and of the residual vectors of the original system of equations are returned.

Reverse communication has the following advantages.

(1) Maximum flexibility in the representation and storage of sparse matrices: all matrix operations are performed outside the solver routine, thereby avoiding the need for a complicated interface with enough flexibility to cope with all types of storage schemes and sparsity patterns. This applies also to preconditioners.

(2) Enhanced user interaction: the progress of the solution can be closely monitored by the user and tidy or immediate termination can be requested. This is useful, for example, when alternative termination criteria are to be employed or in case of failure of the external routines used to perform matrix operations.

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

[3] 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

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

[5] Higham N J (1988) FORTRAN codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation *ACM Trans. Math. Software* **14** 381–396

[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

Note. This routine uses **reverse communication**. Its use involves an initial entry, intermediate exits and re-entries, and a final exit, as indicated by the **parameter IREVCM**. Between intermediate exits and re-entries **all parameters other than IREVCM and V must remain unchanged**.

1: IREVCM — INTEGER *Input/Output*

On initial entry: IREVCM = 0, otherwise an error condition will be raised.

On intermediate re-entry: IREVCM must either be unchanged from its previous exit value, or can have one of the following values.

5 Tidy termination: the computation will terminate at the end of the current iteration. Further reverse communication exits may occur depending on when the termination request is issued. F11BSF will then return with the termination code IREVCM = 4. Note that before calling F11BSF with IREVCM = 5 the calling program must have performed the tasks required by the value of IREVCM returned by the previous call to F11BSF, otherwise subsequently returned values may be invalid.

6 Immediate termination: F11BSF will return immediately with termination code IREVCM = 4 and with any useful information available. This includes the last iterate of the solution. The residual vector is generally not available. F11BSF will then return with the termination code IREVCM = 4.
Immediate termination may be useful, for example, when errors are detected during matrix-vector multiplication or during the solution of the preconditioning equation.

Changing IREVCM to any other value between calls will result in an error.

On intermediate exit: IREVCM has the following meanings.

-1 The calling program must compute the matrix-vector product $v = A^H u$, where u and v are stored in U and V, respectively; RGMRES, CGS and Bi-CGSTAB (ℓ) methods return IREVCM = -1 only if the matrix norm $\|A\|_1$ or $\|A\|_\infty$ is estimated internally using Higham's method. This can only happen if ITERM = 1 in F11BRF.

1 The calling program must compute the matrix-vector product $v = Au$, where u and v are stored in U and V, respectively.

2 The calling program must solve the preconditioning equation $Mv = u$, where u and v are stored in U and V, respectively.

3 Monitoring step: the solution and residual at the current iteration are returned in the arrays U and V, respectively. No action by the calling program is required. F11BTF can be called at this step to return additional information.

On final exit: IREVCM = 4: F11BSF has completed its tasks. The value of IFAIL determines whether the iteration has been successfully completed, errors have been detected or the calling program has requested termination.

Constraints: on initial entry, IREVCM = 0; on re-entry, either IREVCM must remain unchanged or be reset to 5 or 6.

2: U($*$) — ***complex*** array *Input/Output*

Note: the dimension of the array U must be at least n.

On initial entry: an initial estimate, x_0, of the solution of the system of equations $Ax = b$.

On intermediate re-entry: U must remain unchanged.

On intermediate exit: the returned value of IREVCM determines the contents of U in the following way:

IREVCM = -1, 1 or 2	U holds the vector u on which the operation specified by IREVCM is to be carried out;
IREVCM = 3	U holds the current iterate of the solution vector.

On final exit: if IFAIL = 3 or < 0, the array U is unchanged from the initial entry to F11BSF. If IFAIL = 1, the array U is unchanged from the last entry to F11BSF. Otherwise, U holds the last available iterate of the solution of the system of equations, for all returned values of IFAIL.

3: V($*$) — ***complex*** array *Input/Output*

Note: the dimension of the array V must be at least n.

On initial entry: the right-hand side b of the system of equations $Ax = b$.

On intermediate re-entry: the returned value of IREVCM determines the contents of V in the following way:

IREVCM = -1, 1 or 2	V must store the vector v, the result of the operation specified by the value of IREVCM returned by the previous call to F11BSF;
IREVCM = 3	V must remain unchanged.

On intermediate exit: if IREVCM = 3, V holds the current iterate of the residual vector. Note that this is an approximation to the true residual vector. Otherwise, it does not contain any useful information.

On final exit: if IFAIL = 3 or < 0, the array V is unchanged from the initial entry to F11BSF. If IFAIL = 1, the array V is unchanged from the last entry to F11BSF. If IFAIL= 0 or 2, the array V contains the true residual vector of the system of equations (see also Section 6); otherwise, V stores the last available iterate of the residual vector unless IFAIL = 8 is returned on last entry, in which case V is set to 0.0.

4: WGT($*$) — ***real*** array *Input*

Note: the dimension of the array WGT must be at least $\max(1, n)$ if weights are used, 1 otherwise.

On entry: the user-supplied weights, if these are to be used in the computation of the vector norms in the termination criterion (see Sections 3 and 5 of the document for F11BRF).

5: WORK (LWORK) — ***complex*** array *Input/Output*

On initial entry: the workspace WORK as returned by F11BRF (see also Section 5 of the document for F11BRF).

On intermediate re-entry: WORK must remain unchanged.

6: LWORK — INTEGER *Input*

On initial entry: the dimension of the array WORK as declared in the (sub)program from which F11BSF was called (see also Sections 3 and 5 of the document for F11BRF). The required amount of workspace is as follows:

Method	**Requirements**
RGMRES	LWORK $= 120 + n(m+3) + m(m+5) + 1$, where m is the dimension of the basis;
CGS	LWORK $= 120 + 7n$;
Bi-CGSTAB (ℓ)	LWORK $= 120 + (2n + \ell)(\ell + 2) + p$, where ℓ is the order of the method;
TFQMR	LWORK $= 120 + 10n$;

where

$p = 2n$	if $\ell > 1$ and ITERM $= 2$ was supplied;
$p = n$	if $\ell > 1$ and a preconditioner is used or ITERM $= 2$ was supplied;
$p = 0$	otherwise.

Constraint: LWORK $\geq$ LWREQ, where LWREQ is returned by F11BRF.

7: IFAIL — INTEGER *Input*

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 $= -i$

On entry, the ith argument had an illegal value.

IFAIL $= 1$

F11BSF has been called again after returning the termination code IREVCM $= 4$. No further computation has been carried out and all input data and data stored for access by F11BTF have remained unchanged.

IFAIL $= 2$

The required accuracy could not be obtained. However, F11BSF has terminated with reasonable accuracy: the last iterate of the residual satisfied the termination criterion but the exact residual $r = b - Ax$, did not. After the first occurrence of this situation, the iteration was restarted once, but F11BSF could not improve on the accuracy. 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. You should call F11BTF to check the values of the left-and right-hand side of the termination condition.

IFAIL $= 3$

F11BRF was either not called before calling F11BSF or it returned an error. The arguments U and V remain unchanged.

IFAIL = 4

The calling program requested a tidy termination before the solution had converged. The arrays U and V return the last iterates available of the solution and of the residual vector, respectively.

IFAIL = 5

The solution did not converge within the maximum number of iterations allowed. The arrays U and V return the last iterates available of the solution and of the residual vector, respectively.

IFAIL = 6

Algorithmic breakdown. The last available iterates of the solution and residuals are returned, although it is possible that they are completely inaccurate.

IFAIL = 8

The calling program requested an immediate termination. However, the array U returns the last iterate of the solution, the array V returns the last iterate of the residual vector, for the CGS and TFQMR methods only.

7 Accuracy

On completion, i.e., IREVCM = 4 on exit, the arrays U and V will return the solution and residual vectors, x_k and $r_k = b - Ax_k$, respectively, at the kth iteration, the last iteration performed, unless an immediate termination was requested.

On successful completion, the termination criterion is satisfied to within the user-specified tolerance, as described in Section 3 of the document for F11BRF. The computed values of the left- and right-hand sides of the termination criterion selected can be obtained by a call to F11BTF.

8 Further Comments

The number of operations carried out by F11BSF for each iteration is likely to be principally determined by the computation of the matrix-vector products $v = Au$ and by the solution of the preconditioning equation $Mv = u$ in the calling program. Each of these operations is carried out once every iteration.

The number of the remaining operations in F11BSF for each iteration is approximately proportional to n.

The number of iterations required to achieve a prescribed accuracy cannot be easily determined at the onset, as it can depend dramatically on the conditioning and spectrum of the preconditioned matrix of the coefficients $\bar{A} = M^{-1}A$ (RGMRES, CGS and TFQMR methods) or $\bar{A} = AM^{-1}$ (Bi-CGSTAB (ℓ) method).

Additional matrix-vector products are required for the computation of $\|A\|_1$ or $\|A\|_\infty$, when this has not been supplied to F11BRF and is required by the termination criterion employed.

If the termination criterion $\|r_k\|_p \leq \tau\left(\|b\|_p + \|A\|_p \|x_k\|_p\right)$ is used (see Section 3 of the document for F11BRF) and $\|x_0\| \gg \|x_k\|$, then the required accuracy cannot be obtained due to loss of significant digits. The iteration is restarted automatically at some suitable point: F11BSF sets $x_0 = x_k$ and the computation begins again. For particularly badly scaled problems, more than one restart may be necessary. This does not apply to the RGMRES method which, by its own nature, self-restarts every super-iteration. Naturally, restarting adds to computational costs: it is recommended that the iteration should start from a value x_0 which is as close to the true solution $\tilde{x}$ as can be estimated. Otherwise, the iteration should start from $x_0 = 0$.

9 Example

See the example for F11BRF.