NAG Toolbox for MATLAB

f11bt

1 Purpose

f11bt is the third in a suite of three functions for the iterative solution of a complex general (non-Hermitian) system of simultaneous linear equations (see Golub and Van Loan 1996). f11bt returns information about the computations during an iteration and/or after this has been completed. The first function of the suite, f11br, is a setup function; the second function, f11bs, is the iterative solver itself.

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

2 Syntax

```
[itn, stplhs, stprhs, anorm, sigmax, work, ifail] = f11bt(work, 'lwork',
lwork)
```

3 Description

f11bt returns information about the solution process. It can be called either during a monitoring step of f11bs or after f11bs has completed its tasks. Calling f11bt at any other time will result in an error condition being raised.

For further information you should read the documentation for f11br and f11bs.

4 References

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

5.1 Compulsory Input Parameters

1: **work**(**lwork**) – **complex array**

The workspace **work** as returned by f11bs (see also Sections 3 and 5 of the document for f11bs).

5.2 Optional Input Parameters

1: **lwork – int32 scalar**

Default: The dimension of the array **work**.

Constraint: **lwork** ≥ 120.

Note: although the minimum value of **lwork** ensures the correct functioning of f11bt, a larger value is required by the iterative solver f11bs (see also f11br).

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **itn – int32 scalar**

The number of iterations carried out by f11bs.

2: **stplhs – double scalar**

The current value of the left-hand side of the termination criterion used by f11bs.

3: **stprhs – double scalar**

The current value of the right-hand side of the termination criterion used by f11bs.

4: **anorm – double scalar**

The norm $\|A\|_1$ or $\|A\|_\infty$ when it is used by the termination criterion in f11bs, either when it has been supplied to f11br or it has been estimated by f11bs (see also Sections 3 and 5 of the document for f11br); otherwise, **anorm** $= 0.0$ is returned.

5: **sigmax – double scalar**

The current estimate of the largest singular value $\sigma_1(\bar{A})$ of the preconditioned iteration matrix when it is used by the termination criterion in f11bs, either when it has been supplied to f11br or it has been estimated by f11bs (see also Sections 3 and 5 of the document for f11br); otherwise, **sigmax** $= 0.0$ is returned.

6: **work**(**lwork**) **– complex array**

7: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= -i$

If **ifail** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows: 1: **itn**, 2: **stplhs**, 3: **stprhs**, 4: **anorm**, 5: **sigmax**, 6: **work**, 7: **lwork**, 8: **ifail**.

ifail $= 1$

f11bt has been called out of sequence. For example, the last call to f11bs did not return **irevcm** $= 3$ or 4.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

```
method = 'TFQMR   ';
precon = 'P';
n = int32(8);
m = int32(1);
tol = 1e-08;
maxitn = int32(20);
anorm = 0;
sigmax = 0;
```

```
monit = int32(2);
lwork = int32(6000);
nnz=int32(24);
a=complex(zeros(10000,1));
a(1:24)=[complex( 2., 1.),
         complex(-1., 1.),
         complex( 1.,-3.),
         complex( 4., 7.),
         complex(-3., 0.),
         complex( 2., 4.),
         complex(-7.,-5.),
         complex( 2., 1.),
         complex( 3., 2.),
         complex(-4., 2.),
         complex( 0., 1.),
         complex( 5.,-3.),
         complex(-1., 2.),
         complex( 8., 6.),
         complex(-3.,-4.),
         complex(-6.,-2.),
         complex( 5.,-2.),
         complex( 2., 0.),
         complex( 0.,-5.),
         complex(-1., 5.),
         complex( 6., 2.),
         complex(-1., 4.),
         complex( 2., 0.),
         complex( 3., 3.)];
irow=zeros(10000,1,'int32');
irow(1:24) = [int32(1),
              int32(1),
              int32(1),
              int32(2),
              int32(2),
              int32(2),
              int32(3),
              int32(3),
              int32(4),
              int32(4),
              int32(4),
              int32(4),
              int32(5),
              int32(5),
              int32(5),
              int32(6),
              int32(6),
              int32(6),
              int32(7),
              int32(7),
              int32(7),
              int32(8),
              int32(8),
              int32(8)];
icol=zeros(10000,1,'int32');
icol(1:24) = [int32(1),
              int32(4),
              int32(8),
              int32(1),
              int32(2),
              int32(5),
              int32(3),
              int32(6),
              int32(1),
              int32(3),
              int32(4),
              int32(7),
              int32(2),
              int32(5),
              int32(7),
              int32(1),
```

```
              int32(3),
              int32(6),
              int32(3),
              int32(5),
              int32(7),
              int32(2),
              int32(6),
              int32(8)];
ipivp = zeros(n,1,'int32');
ipivq = zeros(n,1,'int32');
irevcm = int32(0);
lfill = int32(0);
dtol = 0;
milu = 'N';
wgt = zeros(8,1);
u = complex(zeros(8,1));
v = [complex(  7., 11.);
     complex(  1., 24.);
     complex(-13.,-18.);
     complex(-10.,  3.);
     complex( 23., 14.);
     complex( 17., -7.);
     complex( 15., -3.);
     complex( -3., 20.)];
[a, irow, icol, ipivp, ipivq, istr, idiag, nnzc, npivm, ifail] = ...
    f11dn(nnz, a, irow, icol, lfill, dtol, milu, ipivp, ipivq);

[lwreq, work, ifail] = ...
       f11br(method, precon, n, m, tol, maxitn, anorm, sigmax, monit,
lwork, 'norm_p', '1');

while (irevcm ~= 4)
  [irevcm, u, v, work, ifail] = f11bs(irevcm, u, v, wgt, work);

  if (irevcm == -1)
    [v, ifail] = f11xn('T', a(1:nnz), irow(1:nnz), icol(1:nnz), 'N', u);
  elseif (irevcm == 1)
    [v, ifail] = f11xn('N', a(1:nnz), irow(1:nnz), icol(1:nnz), 'N', u);
  elseif (irevcm == 2)
     [v, ifail] = f11dp('N', a, irow, icol, ipivp, ipivq, istr, idiag,
'N', u);
  elseif (irevcm == 3)
    [itn, stplhs, stprhs, anorm, sigmax, work, ifail] = f11bt(work);
        fprintf('\nMonitoring at iteration number %d\nresidual norm:
%14.4e\n', itn, stplhs);
    fprintf('\n   Solution Vector\n');
    for i = 1:n
      fprintf('%+16.4e + %+16.4eI\n', real(u(i)), imag(u(i)));
    end
    fprintf('\n   Residual Vector\n');
    for i = 1:n
      fprintf('%+16.4e + %+16.4eI\n', real(v(i)), imag(v(i)));
    end
  end
end
% Get information about the computation
[itn, stplhs, stprhs, anorm, sigmax, work, ifail] = f11bt(work);
fprintf('\nNumber of iterations for convergence: %d\n', itn);
fprintf('Residual norm:                          %14.4e\n', stplhs);
fprintf('Right-hand side of termination criteria: %14.4e\n', stprhs);
fprintf('i-norm of matrix a:                     %14.4e\n', anorm);
fprintf('\n   Solution Vector\n');
for i = 1:n
  fprintf('%+16.4e + %+16.4eI\n', real(u(i)), imag(u(i)));
end
fprintf('\n   Residual Vector\n');
for i = 1:n
  fprintf('%+16.4e + %+16.4eI\n', real(v(i)), imag(v(i)));
end
```