NAG Library Routine Document

F07JHF (DPTRFS)

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

F07JHF (DPTRFS) computes error bounds and refines the solution to a real system of linear equations $AX = B$, where A is an n by n symmetric positive definite tridiagonal matrix and X and B are n by r matrices, using the modified Cholesky factorization returned by F07JDF (DPTTRF) and an initial solution returned by F07JEF (DPTTRS). Iterative refinement is used to reduce the backward error as much as possible.

2 Specification

```
SUBROUTINE F07JHF (N, NRHS, D, E, DF, EF, B, LDB, X, LDX, FERR, BERR,      &
                   WORK, INFO)
INTEGER            N, NRHS, LDB, LDX, INFO
REAL (KIND=nag_wp) D(*), E(*), DF(*), EF(*), B(LDB,*), X(LDX,*),           &
                   FERR(NRHS), BERR(NRHS), WORK(2*N)
```

The routine may be called by its LAPACK name ***dptrfs***.

3 Description

F07JHF (DPTRFS) should normally be preceded by calls to F07JDF (DPTTRF) and F07JEF (DPTTRS). F07JDF (DPTTRF) computes a modified Cholesky factorization of the matrix A as

$$A = LDL^{\mathrm{T}},$$

where L is a unit lower bidiagonal matrix and D is a diagonal matrix, with positive diagonal elements. F07JEF (DPTTRS) then utilizes the factorization to compute a solution, $\hat{X}$, to the required equations. Letting $\hat{x}$ denote a column of $\hat{X}$, F07JHF (DPTRFS) computes a *component-wise backward error*, β, the smallest relative perturbation in each element of A and b such that $\hat{x}$ is the exact solution of a perturbed system

$$(A + E)\hat{x} = b + f, \quad \text{with} \quad |e_{ij}| \leq \beta|a_{ij}|, \quad \text{and} \quad |f_j| \leq \beta|b_j|.$$

The routine also estimates a bound for the *component-wise forward error* in the computed solution defined by $\max|x_i - \hat{x}_i|/\max|\hat{x}_i|$, where x is the corresponding column of the exact solution, X.

Note that the modified Cholesky factorization of A can also be expressed as

$$A = U^{\mathrm{T}}DU,$$

where U is unit upper bidiagonal.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia http://www.netlib.org/lapack/lug

5 Arguments

1: N – INTEGER *Input*

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

Constraint: $\mathrm{N} \geq 0$.

2: NRHS – INTEGER *Input*

On entry: r, the number of right-hand sides, i.e., the number of columns of the matrix B.

Constraint: $\mathrm{NRHS} \geq 0$.

3: D($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array D must be at least $\max(1, \mathrm{N})$.

On entry: must contain the n diagonal elements of the matrix of A.

4: E($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array E must be at least $\max(1, \mathrm{N} - 1)$.

On entry: must contain the $(n - 1)$ subdiagonal elements of the matrix A.

5: DF($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array DF must be at least $\max(1, \mathrm{N})$.

On entry: must contain the n diagonal elements of the diagonal matrix D from the LDL^{T} factorization of A.

6: EF($*$) – REAL (KIND=nag_wp) array *Input*

Note: the dimension of the array EF must be at least $\max(1, \mathrm{N})$.

On entry: must contain the $(n - 1)$ subdiagonal elements of the unit bidiagonal matrix L from the LDL^{T} factorization of A.

7: B(LDB, $*$) – REAL (KIND=nag_wp) array *Input*

Note: the second dimension of the array B must be at least $\max(1, \mathrm{NRHS})$.

On entry: the n by r matrix of right-hand sides B.

8: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F07JHF (DPTRFS) is called.

Constraint: $\mathrm{LDB} \geq \max(1, \mathrm{N})$.

9: X(LDX, $*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array X must be at least $\max(1, \mathrm{NRHS})$.

On entry: the n by r initial solution matrix X.

On exit: the n by r refined solution matrix X.

10: LDX – INTEGER *Input*

On entry: the first dimension of the array X as declared in the (sub)program from which F07JHF (DPTRFS) is called.

Constraint: $\mathrm{LDX} \geq \max(1, \mathrm{N})$.

11: FERR(NRHS) – REAL (KIND=nag_wp) array *Output*

On exit: estimate of the forward error bound for each computed solution vector, such that $\|\hat{x}_j - x_j\|_\infty / \|\hat{x}_j\|_\infty \leq \mathrm{FERR}(j)$, where $\hat{x}_j$ is the jth column of the computed solution returned in the array X and x_j is the corresponding column of the exact solution X. The estimate is almost always a slight overestimate of the true error.

12: BERR(NRHS) – REAL (KIND=nag_wp) array *Output*

On exit: estimate of the component-wise relative backward error of each computed solution vector $\hat{x}_j$ (i.e., the smallest relative change in any element of A or B that makes $\hat{x}_j$ an exact solution).

13: WORK($2 \times \mathrm{N}$) – REAL (KIND=nag_wp) array *Workspace*

14: INFO – INTEGER *Output*

On exit: $\mathrm{INFO} = 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

$\mathrm{INFO} < 0$

If $\mathrm{INFO} = -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed solution for a single right-hand side, $\hat{x}$, satisfies an equation of the form

$$(A + E)\hat{x} = b,$$

where

$$\|E\|_\infty = O(\epsilon)\|A\|_\infty$$

and ϵ is the ***machine precision***. An approximate error bound for the computed solution is given by

$$\frac{\|\hat{x} - x\|_\infty}{\|x\|_\infty} \leq \kappa(A)\frac{\|E\|_\infty}{\|A\|_\infty},$$

where $\kappa(A) = \|A^{-1}\|_\infty \|A\|_\infty$, the condition number of A with respect to the solution of the linear equations. See Section 4.4 of Anderson *et al.* (1999) for further details.

Routine F07JGF (DPTCON) can be used to compute the condition number of A.

8 Parallelism and Performance

F07JHF (DPTRFS) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F07JHF (DPTRFS) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

The total number of floating-point operations required to solve the equations $AX = B$ is proportional to nr. At most five steps of iterative refinement are performed, but usually only one or two steps are required.

The complex analogue of this routine is F07JVF (ZPTRFS).

10 Example

This example solves the equations

$$AX = B,$$

where A is the symmetric positive definite tridiagonal matrix

$$A = \begin{pmatrix} 4.0 & -2.0 & 0 & 0 & 0 \\ -2.0 & 10.0 & -6.0 & 0 & 0 \\ 0 & -6.0 & 29.0 & 15.0 & 0 \\ 0 & 0 & 15.0 & 25.0 & 8.0 \\ 0 & 0 & 0 & 8.0 & 5.0 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 6.0 & 10.0 \\ 9.0 & 4.0 \\ 2.0 & 9.0 \\ 14.0 & 65.0 \\ 7.0 & 23.0 \end{pmatrix}.$$

Estimates for the backward errors and forward errors are also output.

10.1 Program Text

```
    Program f07jhfe

!     F07JHF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: dptrfs, dpttrf, dpttrs, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: i, ifail, info, ldb, ldx, n, nrhs
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: b(:,:), berr(:), d(:), df(:), e(:),  &
                                          ef(:), ferr(:), work(:), x(:,:)
!     .. Executable Statements ..
      Write (nout,*) 'F07JHF Example Program Results'
      Write (nout,*)
      Flush (nout)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, nrhs
      ldb = n
      ldx = n
      Allocate (b(ldb,nrhs),berr(nrhs),d(n),df(n),e(n-1),ef(n-1),ferr(nrhs),   &
        work(2*n),x(ldx,nrhs))

!     Read the lower bidiagonal part of the tridiagonal matrix A from
!     data file

      Read (nin,*) d(1:n)
      Read (nin,*) e(1:n-1)

!     Read the right hand matrix B

      Read (nin,*)(b(i,1:nrhs),i=1,n)

!     Copy A into DF and EF, and copy B into X

      df(1:n) = d(1:n)
```

```
      ef(1:n-1) = e(1:n-1)
      x(1:n,1:nrhs) = b(1:n,1:nrhs)

!     Factorize the copy of the tridiagonal matrix A
!     The NAG name equivalent of dpttrf is f07jdf
      Call dpttrf(n,df,ef,info)

      If (info==0) Then

!       Solve the equations AX = B
!       The NAG name equivalent of dpttrs is f07jef
        Call dpttrs(n,nrhs,df,ef,x,ldx,info)

!       Improve the solution and compute error estimates
!       The NAG name equivalent of dptrfs is f07jhf
        Call dptrfs(n,nrhs,d,e,df,ef,b,ldb,x,ldx,ferr,berr,work,info)

!       Print the solution and the forward and backward error
!       estimates

!       ifail: behaviour on error exit
!              =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
        ifail = 0
        Call x04caf('General',' ',n,nrhs,x,ldx,'Solution(s)',ifail)

        Write (nout,*)
        Write (nout,*) 'Backward errors (machine-dependent)'
        Write (nout,99999) berr(1:nrhs)
        Write (nout,*)
        Write (nout,*) 'Estimated forward error bounds (machine-dependent)'
        Write (nout,99999) ferr(1:nrhs)
      Else
        Write (nout,99998) 'The leading minor of order ', info,                &
          ' is not positive definite'
      End If

99999 Format ((3X,1P,7E11.1))
99998 Format (1X,A,I3,A)
    End Program f07jhfe
```

10.2 Program Data

```
F07JHF Example Program Data
  5     2                      :Values of N and NRHS
  4.0  10.0  29.0  25.0   5.0 :End of diagonal D
 -2.0  -6.0  15.0   8.0       :End of super-diagonal E
  6.0  10.0
  9.0   4.0
  2.0   9.0
 14.0  65.0
  7.0  23.0                    :End of matrix B
```

10.3 Program Results

```
 F07JHF Example Program Results

 Solution(s)
             1          2
 1      2.5000     2.0000
 2      2.0000    -1.0000
 3      1.0000    -3.0000
 4     -1.0000     6.0000
 5      3.0000    -5.0000
```

```
 Backward errors (machine-dependent)
       0.0E+00    7.4E-17

 Estimated forward error bounds (machine-dependent)
       2.4E-14    4.7E-14
```