

F07HEFP (PDPBTRS)

NAG Parallel Library Routine Document

Note: before using this routine, please read the Users' Note for your implementation to check for implementation-dependent details. You are advised to enclose any calls to NAG Parallel Library routines between calls to Z01AAFP and Z01ABFP.

1 Description

F07HEFP (PDPBTRS) solves an n by n real symmetric positive-definite band system of linear equations with multiple right-hand sides, $A_s X = B_s$, where A_s is a submatrix of a larger n by n_A matrix A , i.e.,

$$A_s(1:n, 1:n) \equiv A(1:n, j_A : j_A + n - 1),$$

and B_s is a n by r submatrix of a larger m_B by r matrix B , i.e.,

$$B_s(1:n, 1:r) \equiv B(i_B : i_B + n - 1, 1:r).$$

Note: if $j_A = 1$ and $n = n_A$, then $A_s = A$; if $i_B = 1$ and $n = m_B$, then $B_s = B$.

The band matrix A_s must have been previously factorized by a call to F07HDFP (PDPBTRF), which then performs a Cholesky factorization and F07HEFP (PDPBTRS) then solves the band system of equations by a parallel forward and backward substitution.

2 Specification

```

SUBROUTINE F07HEFP(UPLO, N, BW, NRHS, A, JA, IDESCA, B, IB,
1             IDESCB, AF, LAF, WORK, LWORK, INFO)
ENTRY        PDPBTRS(UPLO, N, BW, NRHS, A, JA, IDESCA, B, IB,
1             IDESCB, AF, LAF, WORK, LWORK, INFO)
INTEGER      N, BW, NRHS, JA, IDESCA(*), IB, IDESCB(*), LAF,
1             LWORK, INFO
DOUBLE PRECISION  A(*), B(*), AF(*), WORK(*)
CHARACTER*1    UPLO

```

The ENTRY statement enables the routine to be called by its ScaLAPACK name.

3 Usage

3.1 Definitions

The following definitions are used in describing the data distribution within this document:

- b_w – the half band width of the symmetric banded matrix;
- m_p – the number of rows in the Library Grid, for this routine $m_p = 1$ or $m_p = p$;
- n_p – the number of columns in the Library Grid, for this routine $n_p = 1$ or $n_p = p$.
- p – $m_p \times n_p$, the total number of processors in the Library Grid.
- M_b^X – the blocking factor for the distribution of the rows of a matrix X .
- N_b^X – the blocking factor for the distribution of the columns of a matrix X .

3.2 Global and Local Arguments

The following global **input** arguments must have the same value on entry to the routine on each processor and the global **output** arguments will have the same value on exit from the routine on each processor:

Global input arguments: UPLO, N, BW, NRHS, JA, IB, some elements of IDESCA and IDESCB (see Section 4 for a description of IDESCA and IDESCB).

Global output arguments: INFO.

The remaining arguments are local.

3.3 Distribution Strategy

The factorized matrix A should be distributed over a one-dimensional array of processors, assuming a column block distribution, as in F07HDFP (PDPBTRF). The right-hand sides of the equation, B_s are stored in the array B in a row block distribution. **It is important that** $p \times N_b^A \geq \text{mod}(j_A - 1, N_b^A) + n$ and $N_b^A \geq 2 \times b_w$, with $M_b^B = N_b^A$ and $i_B = j_A$. This restriction states that the mapping for matrices must be blocked, due to alignment restriction and reflecting the nature of the **divide and conquer algorithm** as a task-parallel algorithm. This means that no processor may store more than one block of the matrix.

3.4 Related Routines

The Library provides many support routines for the generation/distribution and input/output of data in column or row block form. The following routines may be used in conjunction with F07HEFP (PDPBTRS):

Real matrix generation:	column block distribution: F01YXFP
Real matrix generation:	row block distribution: F01YYFP
Real matrix output:	row block distribution: X04BXP

4 Arguments

1: UPLO — CHARACTER*1 *Global Input*

On entry: indicates whether A_s has been factorized as PU^TUP^T or PLL^TP^T as follows:

if UPLO = 'U', then $A_s = PU^TUP^T$, where U is upper triangular and P is a permutation matrix;

if UPLO = 'L', then $A_s = PLL^TP^T$, where L is lower triangular and P is a permutation matrix.

Constraint: UPLO = 'U' or 'L'.

2: N — INTEGER *Global Input*

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

Constraint: $N \geq 0$.

3: BW — INTEGER *Global Input*

On entry: b_w , the number of super-diagonals or sub-diagonals of the matrix A_s .

Constraints: $1 \leq BW \leq N-1$.

Note: F07HEFP is suitable for the solving of 'narrow' banded system of equations. Hence, for large N , BW should be much smaller than $N-1$.

4: NRHS — INTEGER *Global Input*

On entry: r , the number of right-hand sides.

Constraint: NRHS ≥ 1 .

5: A(*) — DOUBLE PRECISION array *Local Input*

Note: A must not be changed between calls to the factorize and the solve routines.

Note: the array A is formally defined as a vector. However, you may find it more convenient to consider A as a two-dimensional array of dimension (ℓ_A, γ) where $\ell_A = \text{IDESCA}(9)$ if $\text{IDESCA}(1)=1$, or $\ell_A = \text{IDESCA}(6)$ if $\text{IDESCA}(1) = 501$; and $\gamma \geq N_b^A$.

On entry: the local part of the distributed matrix A which must contain the information about the factorization of the matrix A_s as returned by F07HDFP (PDPBTRF).

6: JA — INTEGER*Global Input*

On entry: j_A , the column index of matrix A , that identifies the first column of the submatrix A_s factorized by F07HDFP (PDPBTRF).

Constraints: $1 \leq JA \leq n_A - N + 1$.

7: IDESCA(*) — INTEGER array*Local Input*

Note: the dimension of the array IDESCA must be at least 6 when IDESCA(1) = 501 and must be at least 9 when IDESCA(1) = 1.

Distribution: if IDESCA(1) = 1, the array elements IDESCA(3:8) must be global to the processor grid. If IDESCA(1) = 501, then only the array elements IDESCA(3:5) must be global. In either case IDESCA(2) is local to each processor.

On entry: the description array for the matrix A . This array must contain details of the distribution of the matrix A and the logical processor grid.

IDESCA(1), the descriptor type.

If IDESCA(1) = 1, then $p = 1 \times n_p$ and:

IDESCA(2), the Library context, usually returned by a call to the Library Grid initialisation routine Z01AAFP;

IDESCA(3), the number of rows, m_A , of the matrix A ;

IDESCA(4), the number of columns, n_A , of the matrix A ;

IDESCA(5), the blocking factor, M_b^A , used to distribute the rows of the matrix A ;

IDESCA(6), the blocking factor, N_b^A , used to distribute the columns of the matrix A ;

IDESCA(7), the processor row index over which the first row of the matrix A is distributed;

IDESCA(8), the processor column index over which the first column of the matrix A is distributed;

IDESCA(9), the leading dimension of the conceptual two-dimensional array A .

If IDESCA(1) = 501, then $p = 1 \times n_p$ or $p = m_p \times 1$, and:

IDESCA(2), the Library context, usually returned by a call to the Library Grid initialisation routine Z01AAFP;

IDESCA(3), the size n_A , of the matrix A ;

IDESCA(4), the blocking factor, N_b^A , used to distribute the matrix A ;

IDESCA(5), the processor column index over which the first element of the matrix A is distributed;

IDESCA(6), the leading dimension of the conceptual two-dimensional array A .

IDESCA(7:9) are not referenced.

Suggested value: IDESCA(1) = 501 and $p = 1 \times n_p$.

Constraints:

IDESCA(1) = 1 or 501;

if IDESCA(1) = 1, then $p = 1 \times n_p$;

if IDESCA(1) = 501; then $p = m_p \times 1$ or $p = 1 \times n_p$;

if IDESCA(1) = 501, then

$1 \leq \text{IDESCA}(3) \leq N + \text{JA} - 1$;

$\text{IDESCA}(4) \geq 2 \times \text{BW}$ and $p \times \text{IDESCA}(4) \geq \text{mod}(\text{JA} - 1, \text{IDESCA}(4)) + N$;

$\text{IDESCA}(5) \geq 0$;

$\text{IDESCA}(6) \geq \text{BW} + 1$;

if IDESCA(1) = 1, then

$1 \leq \text{IDESCA}(4) \leq N + \text{JA} - 1$;

$\text{IDESCA}(6) \geq 2 \times \text{BW}$ and $p \times \text{IDESCA}(6) \geq \text{mod}(\text{JA} - 1, \text{IDESCA}(6)) + N$;

$\text{IDESCA}(8) \geq 0$;

$\text{IDESCA}(9) \geq \text{BW} + 1$.

8: B(*) — DOUBLE PRECISION array *Local Input/Local Output*

Note: the array B is formally defined as a vector. However, you may find it more convenient to consider B as a two-dimensional array of dimension (ℓ_B, γ) where $\ell_B = \text{IDESCB}(9)$ if $\text{IDESCB}(1) = 1$, or $\ell_B = \text{IDESCB}(6)$ if $\text{IDESCB}(1) = 502$; and $\gamma \geq r$.

On entry: the local part of the right-hand side B which is stored in row block fashion.

On exit: the n by r solution matrix X distributed in the same row block distribution.

9: IB — INTEGER *Global Input*

On entry: i_B , the row index of the matrix B that identifies the first row of the submatrix B_s .

Constraints: $1 \leq \text{IB}$ and $\text{IB} = \text{JA}$.

10: IDESCB(*) — INTEGER array *Local Input*

Note: the dimension of the array IDESCB must be at least 6 when $\text{IDESCB}(1) = 502$ and must be at least 9 when $\text{IDESCB}(1) = 1$.

Distribution: if $\text{IDESCB}(1) = 1$, the array elements $\text{IDESCB}(3:8)$ must be global to the processor grid. If $\text{IDESCB}(1) = 502$, then only the array elements $\text{IDESCB}(3:5)$ must be global. In either case $\text{IDESCB}(2)$ is local to each processor.

On entry: the description array for the matrix B . This array must contain details of the distribution of the matrix B and the logical processor grid.

IDESCB(1), the descriptor type.

If $\text{IDESCB}(1) = 1$, then $p = m_p \times 1$ and:

IDESCB(2), the Library context, usually returned by a call to the Library Grid initialisation routine Z01AAFP;

IDESCB(3), the number of rows, m_B , of the matrix B ;

IDESCB(4), the number of columns, n_B , of the matrix B ;

IDESCB(5), the blocking factor, M_b^B , used to distribute the rows of the matrix B ;

IDESCB(6), the blocking factor, N_b^B , used to distribute the columns of the matrix B ;

IDESCB(7), the processor row index over which the first row of the matrix B is distributed;

IDESCB(8), the processor column index over which the first column of the matrix B is distributed;

IDESCB(9), the leading dimension of the conceptual two-dimensional array B.

If $\text{IDESCB}(1) = 502$, then $p = 1 \times n_p$, or $p = m_p \times 1$, and:

IDESCB(2), the Library context, usually returned by a call to the Library Grid initialisation routine Z01AAFP;

IDESCB(3), the size m_B , of the matrix B ;

IDESCB(4), the blocking factor, M_b^B , used to distribute the matrix B ;

IDESCB(5), the processor column index over which the first element of the matrix B is distributed;

IDESCB(6), the leading dimension of the conceptual two-dimensional array B;

IDESCB(7:9) are not referenced.

Suggested value: $\text{IDESCB}(1) = 502$ and $p = 1 \times n_p$.

Constraints:

IDESCB(1) = 1 or 502;

if $\text{IDESCB}(1) = 1$ or 502, then $p = 1 \times n_p$;

if $\text{IDESCB}(1) = 502$, then $p = m_p \times 1$;

if $\text{IDESCB}(1) = 502$, then

$1 \leq \text{IDESCB}(3) \leq N + \text{IB} - 1$;

$\text{IDESCB}(4) \geq 2 \times \text{BW}$ and $p \times \text{IDESCB}(4) \geq \text{mod}(\text{IB} - 1, \text{IDESCB}(4)) + N$;

$\text{IDESCB}(5) \geq 0$;
 $\text{IDESCB}(6) \geq \text{BW}+1$;
 $\text{IDESCB}(2) = \text{IDESCA}(2)$;
 and if $\text{IDESCA}(1) = 501$, then
 $\text{IDESCB}(4) = \text{IDESCA}(4)$;
 $\text{IDESCB}(5) = \text{IDESCA}(5)$;
 and if $\text{IDESCA}(1) = 1$, then
 $\text{IDESCB}(4) = \text{IDESCA}(6)$;
 $\text{IDESCB}(5) = \text{IDESCA}(8)$;
 if $\text{IDESCB}(1) = 1$, then
 $1 \leq \text{IDESCB}(4) \leq N + \text{IB}-1$;
 $\text{IDESCB}(6) \geq 2 \times \text{BW}$ and $p \times \text{IDESCB}(6) \geq \text{mod}(\text{IB}-1, \text{IDESCB}(6)) + N$;
 $\text{IDESCB}(8) \geq 0$;
 $\text{IDESCB}(9) \geq \text{BW}+1$;
 $\text{IDESCB}(2) = \text{IDESCA}(2)$;
 and if $\text{IDESCA}(1) = 1$, then
 $\text{IDESCB}(6) = \text{IDESCA}(6)$;
 $\text{IDESCB}(8) = \text{IDESCA}(8)$;
 and if $\text{IDESCA}(1) = 501$, then
 $\text{IDESCB}(6) = \text{IDESCA}(4)$;
 $\text{IDESCB}(8) = \text{IDESCA}(5)$.

11: AF(*) — DOUBLE PRECISION array *Local Input*

Note: AF **must not be changed** between calls to the factorize and the solve routines.

On entry: the auxiliary fill-in space, created and stored by F07HDFP (PDPBTRF). If LAF is not large enough, after an unsuccessful exit, INT(AF(1)) will contain the minimum acceptable size of AF.

12: LAF — INTEGER *Local Input*

On entry: the dimension of the array AF as specified in F07HDFP (PDPBTRF).

Constraint: $\text{LAF} \geq (N_6^A + 2 \times \text{BW}) \times \text{BW}$.

13: WORK(*) — DOUBLE PRECISION array *Local Workspace*

Note: the dimension of WORK must be at least $\max(1, \text{LWORK})$. The minimum value of LWORK required to successfully call this routine can be obtained by setting $\text{LWORK} = -1$. The required size is returned in array element WORK(1).

On exit: WORK(1) contains the minimum dimension of the array WORK required to successfully complete the task.

14: LWORK — INTEGER *Local Input*

On entry: either -1 (see WORK) or the dimension of the array WORK required to successfully complete the task. If LWORK is set to -1 on entry this routine simply performs some initial error checking and then, if these checks are successful, calculates the minimum size of LWORK required.

Constraints:

either $\text{LWORK} = -1$,
 or $\text{LWORK} \geq \text{NRHS} \times \text{BW}$.

15: INFO — INTEGER *Global Output*

The NAG Parallel Library provides a mechanism, via the routine Z02EAFP, to reduce the amount of parameter validation performed by this routine. For a full description refer to the Z02 Chapter Introduction.

On exit: INFO = 0 (or -9999 if reduced error checking is enabled) unless the routine detects an error (see Section 5).

5 Errors and Warnings

If $\text{INFO} \neq 0$ explanatory error messages are output from the root processor (or processor $\{0,0\}$ when the root processor is not available) on the current error message unit (as defined by X04AAF).

$\text{INFO} = -i$

On entry, one of the arguments was invalid:

if the k th argument is a scalar $\text{INFO} = -k$;

if the k th argument is an array and its j th element is invalid, $\text{INFO} = -(100 \times k + j)$.

This error occurred either because a global argument did not have the same value on all logical processors, or because its value on one or more processors was incorrect.

6 Further Comments

The total number of floating-point operations is approximately $4b_w nr$, assuming $n \gg b_w$.

6.1 Algorithmic Detail

Forward and backward substitution is used.

If $\text{UPLO} = \text{'U'}$, $A = PU^TUP^T$, where U is band upper triangular and P is a permutation matrix; the solution X is computed by solving $PU^TY = B$ and then $UP^TX = Y$.

If $\text{UPLO} = \text{'L'}$, $A = PLL^TP^T$, where L is band lower triangular and P is a permutation matrix; the solution X is computed by solving $PLY = B$ and then $L^TP^TX = Y$.

6.2 Parallelism Detail

The Level-3 BLAS operations used in this routine are carried out in parallel.

6.3 Accuracy

For each right-hand side vector b , the computed solution x is the exact solution of a perturbed system of equations $(A + E)x = b$, where

$$|E| \leq c(b_w + 1)\epsilon|U^T| \cdot |U| \quad \text{if UPLO} = \text{'U'},$$

$$|E| \leq c(b_w + 1)\epsilon|L| \cdot |L^T| \quad \text{if UPLO} = \text{'L'},$$

$c(b_w + 1)$ is a modest linear function of $b_w + 1$ and ϵ is the *machine precision*. If x is the true solution, then the computed solution \hat{x} satisfies a forward error bound of the form

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq \epsilon c(b_w + 1)\kappa(A),$$

where $\kappa(A)$ is the condition number of A . See the F07 Chapter Introduction.

7 References

- [1] Blackford L S, Choi J, Cleary A, D'Azevedo E, Demmel J, Dhillon I, Dongarra J, Hammarling S, Henry G, Petitet A, Stanley K, Walker D and Whaley R C (1997) *ScaLAPACK Users' Guide* SIAM 3600 University City Science Center, Philadelphia, PA 19104-2688, USA. URL: <http://www.netlib.org/scalapack/slug/scalapack.slug.html>
- [2] 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
- [3] Golub G H and van Loan C F (1996) *Matrix Computations* Johns Hopkins University Press (3rd Edition), Baltimore

8 Example

See Section 8 of the document for F07HDFP (PDPBTRF).
