# NAG Library Function Document nag\_dpbstf (f08ufc)

## 1 Purpose

nag\_dpbstf (f08ufc) computes a split Cholesky factorization of a real symmetric positive definite band matrix.

# 2 Specification

# 3 Description

nag\_dpbstf (f08ufc) computes a split Cholesky factorization of a real symmetric positive definite band matrix B. It is designed to be used in conjunction with nag dsbgst (f08uec).

The factorization has the form  $B = S^T S$ , where S is a band matrix of the same bandwidth as B and the following structure: S is upper triangular in the first (n+k)/2 rows, and transposed — hence, lower triangular — in the remaining rows. For example, if n=9 and k=2, then

$$S = \begin{pmatrix} s_{11} & s_{12} & s_{13} \\ & s_{22} & s_{23} & s_{24} \\ & & s_{33} & s_{34} & s_{35} \\ & & & s_{44} & s_{45} \\ & & & s_{55} \\ & & & s_{64} & s_{65} & s_{66} \\ & & & s_{75} & s_{76} & s_{77} \\ & & & & s_{86} & s_{87} & s_{88} \\ & & & & s_{97} & s_{98} & s_{99} \end{pmatrix}$$

# 4 References

None.

#### 5 Arguments

1: **order** – Nag OrderType

Input

On entry: the **order** argument specifies the two-dimensional storage scheme being used, i.e., row-major ordering or column-major ordering. C language defined storage is specified by **order** = Nag\_RowMajor. See Section 3.2.1.3 in the Essential Introduction for a more detailed explanation of the use of this argument.

Constraint: order = Nag\_RowMajor or Nag\_ColMajor.

2: **uplo** – Nag\_UploType

Input

On entry: indicates whether the upper or lower triangular part of B is stored.

uplo = Nag\_Upper

The upper triangular part of B is stored.

Mark 24 f08ufc.1

**uplo** = Nag\_Lower

The lower triangular part of B is stored.

Constraint: uplo = Nag\_Upper or Nag\_Lower.

3:  $\mathbf{n}$  - Integer Input

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

Constraint:  $\mathbf{n} \geq 0$ .

4: **kb** – Integer Input

On entry: if **uplo** = Nag\_Upper, the number of superdiagonals,  $k_b$ , of the matrix B.

If **uplo** = Nag\_Lower, the number of subdiagonals,  $k_b$ , of the matrix B.

Constraint:  $\mathbf{kb} \geq 0$ .

5:  $\mathbf{bb}[dim]$  - double Input/Output

**Note**: the dimension, dim, of the array **bb** must be at least  $max(1, pdbb \times n)$ .

On entry: the n by n symmetric positive definite band matrix B.

This is stored as a notional two-dimensional array with row elements or column elements stored contiguously. The storage of elements of  $B_{ij}$ , depends on the **order** and **uplo** arguments as follows:

```
if \mathbf{order} = \text{'Nag\_ColMajor'} and \mathbf{uplo} = \text{'Nag\_Upper'}, B_{ij} is stored in \mathbf{bb}[k_b+i-j+(j-1)\times\mathbf{pdbb}], for j=1,\ldots,n and i=\max(1,j-k_b),\ldots,j; if \mathbf{order} = \text{'Nag\_ColMajor'} and \mathbf{uplo} = \text{'Nag\_Lower'},
```

 $B_{ij}$  is stored in  $\mathbf{bb}[i-j+(j-1)\times\mathbf{pdbb}]$ , for  $j=1,\ldots,n$  and  $i=j,\ldots,\min(n,j+k_b)$ ;

if order = 'Nag\_RowMajor' and uplo = 'Nag\_Upper',

 $B_{ij}$  is stored in  $\mathbf{bb}[j-i+(i-1)\times\mathbf{pdbb}]$ , for  $i=1,\ldots,n$  and  $j=i,\ldots,\min(n,i+k_b)$ ;

if order = 'Nag\_RowMajor' and uplo = 'Nag\_Lower',

 $B_{ij}$  is stored in  $\mathbf{bb}[k_b+j-i+(i-1)\times\mathbf{pdbb}]$ , for  $i=1,\ldots,n$  and  $j=\max(1,i-k_b),\ldots,i$ .

On exit: B is overwritten by the elements of its split Cholesky factor S.

6: **pdbb** – Integer Input

On entry: the stride separating row or column elements (depending on the value of **order**) of the matrix B in the array **bb**.

*Constraint*:  $\mathbf{pdbb} \ge \mathbf{kb} + 1$ .

7: **fail** – NagError \* Input/Output

The NAG error argument (see Section 3.6 in the Essential Introduction).

## 6 Error Indicators and Warnings

#### **NE BAD PARAM**

On entry, argument  $\langle value \rangle$  had an illegal value.

## NE\_INT

On entry,  $\mathbf{kb} = \langle value \rangle$ . Constraint:  $\mathbf{kb} \geq 0$ .

f08ufc.2 Mark 24

```
On entry, \mathbf{n} = \langle value \rangle.
Constraint: \mathbf{n} \geq 0.
On entry, \mathbf{pdbb} = \langle value \rangle.
Constraint: \mathbf{pdbb} > 0.
```

#### NE INT 2

```
On entry, \mathbf{pdbb} = \langle value \rangle and \mathbf{kb} = \langle value \rangle.
Constraint: \mathbf{pdbb} \ge \mathbf{kb} + 1.
```

#### **NE\_INTERNAL\_ERROR**

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.

## NE\_POS\_DEF

The factorization could not be completed, because the updated element  $b(\langle value \rangle, \langle value \rangle)$  would be the square root of a negative number. Hence B is not positive definite. This may indicate an error in forming the matrix B.

# 7 Accuracy

The computed factor S is the exact factor of a perturbed matrix (B+E), where

$$|E| \le c(k+1)\epsilon |S^{\mathsf{T}}||S|,$$

c(k+1) is a modest linear function of k+1, and  $\epsilon$  is the **machine precision**. It follows that  $|e_{ij}| \leq c(k+1)\epsilon\sqrt{(b_{ii}b_{jj})}$ .

#### 8 Parallelism and Performance

nag dpbstf (f08ufc) is not threaded by NAG in any implementation.

nag\_dpbstf (f08ufc) 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 Users' Note for your implementation for any additional implementation-specific information.

#### **9** Further Comments

The total number of floating-point operations is approximately  $n(k+1)^2$ , assuming  $n \gg k$ .

A call to nag\_dpbstf (f08ufc) may be followed by a call to nag\_dsbgst (f08uec) to solve the generalized eigenproblem  $Az = \lambda Bz$ , where A and B are banded and B is positive definite.

The complex analogue of this function is nag zpbstf (f08utc).

#### 10 Example

See Section 10 in nag\_dsbgst (f08uec).

Mark 24 f08ufc.3 (last)