NAG Fortran Library Routine Document

D06CBF

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

D06CBF generates the sparsity pattern of a finite element matrix associated with a given mesh.

2 Specification

```
SUBROUTINE D06CBF(NV, NELT, NNZMAX, CONN, NNZ, IROW, ICOL, IFAIL)
INTEGER           NV, NELT, NNZMAX, CONN(3,NELT), NNZ, IROW(NNZMAX),
1                 ICOL(NNZMAX), IFAIL
```

3 Description

D06CBF generates the sparsity pattern of a Finite Element matrix associated with a given mesh. The sparsity pattern is returned in a coordinate storage format consistent with the sparse linear algebra routines in the F11 Chapter Introduction. More precisely D06CBF returns the number of non-zero elements in the associated sparse matrix, and their row and column indices. This is designed to assist the user in applying finite element discretisation to meshes from the D06 Chapter Introduction and in solving the resulting sparse linear system using routines from Chapter F11.

The output sparsity pattern is based on the fact that finite element matrix A has elements a_{ij} satisfying:

$$a_{ij} \neq 0 \Longrightarrow i \text{ and } j \text{ are vertices belonging to the same triangle.}$$

4 References

None.

5 Parameters

1: NV – INTEGER *Input*

On entry: the total number of vertices in the input mesh.

Constraint: NV ≥ 3.

2: NELT – INTEGER *Input*

On entry: the number of triangles in the input mesh.

Constraint: NELT $\leq 2 \times$ NV $- 1$.

3: NNZMAX – INTEGER *Input*

On entry: the maximum number of non-zero entries in the matrix based on the input mesh. It is the dimension of the arrays IROW and ICOL as declared in the (sub)program from which D06CBF is called.

Constraint: $4 \times$ NELT $+$ NV $\leq$ NNZMAX $\leq$ NV^2.

4: CONN(3,NELT) – INTEGER array *Input*

On entry: the triangle/vertex connectivity of the mesh. For each triangle j, CONN(i, j) gives the indices of its three vertices (in anticlockwise order) in the all mesh vertices numbering, for $i = 1, 2, 3$ and $j = 1, \ldots,$ NELT.

Constraints:

$1 \leq \text{CONN}(i,j) \leq \text{NV}$,
$\text{CONN}(1,j) \neq \text{CONN}(2,j)$,
$\text{CONN}(1,j) \neq \text{CONN}(3,j)$ and $\text{CONN}(2,j) \neq \text{CONN}(3,j)$, for $i = 1,2,3$ and $j = 1,\ldots,\text{NELT}$.

5: NNZ – INTEGER *Output*

On exit: the number of non-zero entries in the matrix associated with the input mesh.

6: IROW(NNZMAX) – INTEGER array *Output*
7: ICOL(NNZMAX) – INTEGER array *Output*

On exit: the first NNZ elements contain the row and column indices of the non-zero elements supplied in the Finite Element matrix A.

8: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators 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 or warnings detected by the routine:

IFAIL $= 1$

On entry, NV < 3,
or NELT $> 2 \times$ NV $- 1$,
or NNZMAX < 1 or NNZMAX $>$ NV^2
or $\text{CONN}(i,j) < 1$ or $\text{CONN}(i,j) >$ NV for some $i = 1,3$ and j, $1 \leq j \leq$ NELT,
or $\text{CONN}(1,j) = \text{CONN}(2,j)$ or $\text{CONN}(1,j) = \text{CONN}(3,j)$ or
$\text{CONN}(2,j) = \text{CONN}(3,j)$ for some $j = 1,\ldots,\text{NELT}$.

IFAIL $= 2$

A serious error has occurred in an internal call to an auxiliary routine. Check the input mesh, especially the connectivity between triangles and vertices (the argument CONN). Array dimensions should be checked as well. If the problem persists, contact NAG.

7 Accuracy

Not applicable.

8 Further Comments

Not applicable.

9 Example

See Section 9 of the document for D06CCF.