D02NUF – NAG Fortran Library Routine Document

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

D02NUF is a setup routine which must be called by the user, prior to an integrator in the subchapter D02M–D02N, if sparse matrix linear algebra is required.

2 Specification

```
 SUBROUTINE D02NUF(NEQ, NEQMAX, JCEVAL, NWKJAC, IA, NIA, JA, NJA,
1                  JACPVT, NJCPVT, SENS, U, ETA, LBLOCK, ISPLIT,
2                  RWORK, IFAIL)
 INTEGER           NEQ, NEQMAX, NWKJAC, IA(NIA), NIA, JA(NJA), NJA,
1                  JACPVT(NJCPVT), NJCPVT, ISPLIT, IFAIL
 real              SENS, U, ETA, RWORK(50+4*NEQMAX)
 LOGICAL           LBLOCK
 CHARACTER*1       JCEVAL
```

3 Description

This routine defines the linear algebra to be used as sparse matrix linear algebra, permits the user to specify the method for calculating the Jacobian and its structure, and checks the validity of certain input values.

4 References

None.

5 Parameters

1: NEQ — INTEGER *Input*

On entry: the number of differential equations.

Constraint: $1 \leq \text{NEQ} \leq \text{NEQMAX}$.

2: NEQMAX — INTEGER *Input*

On entry: a bound on the maximum number of differential equations to be solved during the integration.

Constraint: $\text{NEQ} \leq \text{NEQMAX}$.

3: JCEVAL — CHARACTER*1 *Input*

On entry: specifies the technique to be used to compute the Jacobian, as follows:

JCEVAL = 'N'

the sparsity structure and the value of the Jacobian are to be determined numerically by the integrator.

JCEVAL = 'S'

the sparsity structure of the Jacobian is supplied in the arrays IA and JA but its value is to be determined numerically. This is the recommended mode of operation unless it is a simple matter to supply the Jacobian.

JCEVAL = 'A'

the Jacobian will be evaluated by calls to a subroutine JAC supplied by the user. The sparsity structure will be estimated by calls to JAC; that is, no explicit sparsity structure need be supplied in the arrays IA and JA.

JCEVAL = 'F'

the sparsity structure of the Jacobian is supplied in IA and JA, and its value will be determined by calls to a subroutine JAC supplied by the user. This is the recommended mode of operation if the subroutine JAC is simple to form.

JCEVAL = 'D'

the default choice is to be made. In this case 'D' is interpreted as 'S'.

If the sparsity structure is supplied in arrays IA and JA, then any evidence from the numerical or analytical formation of the Jacobian that this structure is not correct, is ignored.

Only the first character of the actual argument JCEVAL is passed to D02NUF; hence it is permissible for the actual argument to be more descriptive e.g., 'Numerical', 'Structural', 'Analytical', 'Full information' or 'Default' in a call to D02NUF.

If the option JCEVAL = 'N', 'S' or 'D' is used then the actual argument corresponding to JAC in the call to D02NDF or D02NJF must be either D02NDZ or D02NJZ respectively.

Constraint: JCEVAL = 'N', 'A', 'F' or 'D'.

4: NWKJAC — INTEGER *Input*

On entry: the size of the array WKJAC, that the user is supplying to the integrator, as declared in the (sub)program from which D02NUF is called.

Suggested value: NWKJAC = 4× NEQMAX if JCEVAL = 'N' or 'A'. If NWKJAC is less than this estimate, then a message is printed on the current advisory message unit (see X04ABF), and execution continues.

Constraints: NWKJAC ≥ NELEMENT+2× NEQ if JCEVAL = 'S', 'F' or 'D', where NELEMENT is the total number of non-zeros.

5: IA(NIA) — INTEGER array *Input*

On entry: if JCEVAL = 'S', 'F' or 'D', IA must contain details of the sparsity pattern to be used for the Jacobian. See JA below.

IA is not used if JCEVAL = 'N' or 'A'.

6: NIA — INTEGER *Input*

On entry: the dimension of the array IA as declared in the (sub)program from which D02NUF is called.

Constraints:

NIA ≥ NEQ +1 if JCEVAL = 'S', 'F' or 'D',
NIA ≥ 1 otherwise.

7: JA(NJA) — INTEGER array *Input*

On entry: if JCEVAL = 'S' or 'F' or 'D', JA must contain details of the sparsity pattern to be used for the Jacobian. JA contains the row indices where non-zero elements occur, reading in columnwise order, and IA contains the starting locations in JA of the descriptions of columns 1,2,...,NEQ in that order, with IA(1) = 1. Thus for each column index $j = 1, 2, \ldots,$ NEQ, the values of the row index i in column j where a non-zero element may occur are given by

$$i = \mathrm{JA}(k) \text{ where } \mathrm{IA}(j) \leq k < \mathrm{IA}(j+1).$$

Thus the total number of non-zeros, NELEMENT, must be IA(NEQ + 1) − 1. For example, for the following matrix

$$\begin{pmatrix} x & 0 & x & 0 & 0 \\ 0 & x & x & x & 0 \\ x & x & x & 0 & 0 \\ x & 0 & 0 & x & x \\ 0 & 0 & 0 & x & x \end{pmatrix}$$

where x represents non-zero elements (13 in all) the arrays IA and JA should be

IA(k)	1	4	6	9	12	14							
JA(k)	1	3	4	2	3	1	2	3	2	4	5	4	5

JA is not used if JCEVAL = 'N' or 'A'.

8: NJA — INTEGER *Input*

On entry: the dimension of the array JA as declared in the (sub)program from which D02NUF is called.

Constraints:

NJA ≥ IA(NEQ + 1) − 1 if JCEVAL = 'S', 'F' or 'D',
NJA ≥ 1 otherwise.

9: JACPVT(NJCPVT) — INTEGER array *Workspace*

This must be the same array JACPVT as supplied to the integrator. It is used to pass information about the supplied sparsity structure to the integrator and therefore the contents of this array must not be changed before calling the integrator.

10: NJCPVT — INTEGER *Input*

On entry: the length of the array JACPVT, which the user is supplying to the integrator, as dimensioned in the sub(program) from which D02NUF is called.

Suggested value: NJCPVT = 20× NEQMAX if JCEVAL = 'N' or 'A'. If NJCPVT is less than this estimate, then a message is printed on the current advisory message unit (see X04ABF), and execution continues.

Constraints: NJCPVT ≥ 3× NELEMENT +14× NEQ if JCEVAL = 'S', 'F' or 'D', where NELEMENT is the total number of non-zeros.

11: SENS — ***real*** *Input*

On entry: a threshold parameter used to determine whether or not a matrix element is zero; when SENS is set to 0.0 on entry, the routine will use SENS = 100.0×***machine precision***. Otherwise the absolute value of SENS is used.

12: U — ***real*** *Input*

On entry: U should have a value between 0.0 and 0.9999. Otherwise a default value of 0.1 is used. When the sparsity pattern has been evaluated, the first Jacobian computed is decomposed with U governing the choice of pivots; subsequent Jacobian decompositions use the same pattern of decomposition until the sparsity pattern is re-evaluated. When searching a row for a pivot, any element is excluded from the search which is less than U times the largest of those elements in the row available as pivots. Thus decreasing U biases the algorithm towards maintaining sparsity at the expense of numerical stability.

13: ETA — ***real*** *Input*

On entry: a relative pivot threshold, below which on subsequent decompositions (as described under U above), an internal error is provoked. If ETA > 1.0 then no check on pivot size is made. If ETA ≤ 0.0 then the default value ETA = 1.0E−4 is used.

14: LBLOCK — LOGICAL *Input*

On entry: indicates if preordering is used before decomposition.

If LBLOCK = .TRUE., on entry, the Jacobian matrix is preordered to block lower triangular form before a decomposition is performed (this is the recommended mode). If the user knows the structure of the Jacobian to be irreducible, that is not permutable to block lower triangular form, then the user should set LBLOCK = .FALSE.. For example, a Jacobian arising from using the method of lines for parabolic partial differential equations would normally be irreducible. (See the specification of D02NXF for optional output concerning LBLOCK.)

15: ISPLIT — INTEGER *Input*

On entry: this parameter is used for splitting the integer workspace JACPVT to effect an efficient decomposition. It must satisfy $1 \leq \text{ISPLIT} \leq 99$. If ISPLIT lies outside this range on entry, a default value of 73 is used. An appropriate value for ISPLIT for subsequent runs on similar problems is available via the optional output D02NXF.

Suggested value: ISPLIT = 73, unless the user has information from a previous run of a similar problem.

16: RWORK(50+4∗NEQMAX) — ***real*** array *Workspace*

This must be the same workspace array as the array RWORK supplied to the integrator. It is used to pass information from the setup routine to the integrator and therefore the contents of this array must not be changed before calling the integrator.

17: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

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

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 detected by the routine:

IFAIL = 1

On entry, an illegal input was detected.

7 Accuracy

Not applicable.

8 Further Comments

This routine must be called as a setup routine before a call to either D02NDF or D02NJF and may be called as the linear algebra setup routine before a call to D02NMF or D02NNF.

9 Example

See the example for Section 9 of the document for D02NDF, Section 9 of the document for D02NJF and Section 9 of the document for D02NNF.