NAG Toolbox for MATLAB

d02mv

1 Purpose

d02mv is a setup function which must be called prior to an integrator in sub-chapter D02M/N, if the DASSL implementation of Backward Differentiation Formulae (BDF) is to be used.

2 Syntax

```
[con, rwork, ifail] = d02mv(ldysav, sdysav, maxord, con, tcrit, hmin, hmax, h0, maxstp, mxhnil, norm_p, rwork)
```

3 Description

An integrator setup function must be called before the call to any integrator in this sub-chapter. This setup function d02mv makes the choice of the DASSL integrator and permits you to define options appropriate to this choice.

4 References

See the D02M/N Sub-chapter Introduction.

5 Parameters

5.1 Compulsory Input Parameters

1: **ldysav – int32 scalar**

A bound on the maximum number of differential equations to be solved.

Constraint: **ldysav** ≥ 1.

2: **sdysav – int32 scalar**

The second dimension of the array **ysav** that will be supplied to the integrator, as declared in the (sub)program from which the integrator is called (e.g., see d02nb).

Constraint: **sdysav** $\geq$ **maxord** $+ 3$.

3: **maxord – int32 scalar**

The maximum order to be used for the BDF method. If **maxord** $= 0$ then **maxord** $= 5$ is assumed.

Constraint: $0 \leq$ **maxord** ≤ 5.

4: **con**(3) **– double array**

Values to be used to control step size choice during integration. If any **con**$(i) = 0.0$ on entry, it is replaced by its default value described below. In most cases this is the recommended setting.

con(1), **con**(2), and **con**(3) are factors used to bound step size changes. If the current step size h fails, then the modulus of the next step size is bounded by **con**$(l) \times |h|$. The default value of **con**(1) is 2.0. Note that the new step size may be used with a method of different order to the failed step. If the initial step size is h, then the modulus of the step size on the second step is bounded by **con**$(3) \times |h|$. At any other stage in the integration, if the current step size is h, then the modulus of the next step size is bounded by **con**$(2) \times |h|$. The default values are 10.0 for **con**(2) and 1000.0 for **con**(3).

Constraints:

$0.0 < \mathbf{con}(1) < \mathbf{con}(2) < \mathbf{con}(3)$;
$\mathbf{con}(2) > 1.0$;
$\mathbf{con}(3) > 1.0$.

These constraints must be satisfied after any zero values have been replaced by default values.

5: **tcrit – double scalar**

A point beyond which integration must not be attempted. The use of **tcrit** is described under the parameter **itask** in the specification for the integrator (e.g., see d02nb). A value, 0.0 say, must be specified even if **itask** subsequently specifies that **tcrit** will not be used.

6: **hmin – double scalar**

The minimum absolute step size to be allowed. Set **hmin** $= 0.0$ if this option is not required.

7: **hmax – double scalar**

The maximum absolute step size to be allowed. Set **hmax** $= 0.0$ if this option is not required.

8: **h0 – double scalar**

The step size to be attempted on the first step. Set **h0** $= 0.0$ if the initial step size is to be calculated internally.

9: **maxstp – int32 scalar**

The maximum number of steps to be attempted during one call to the integrator after which it will return with **ifail** $= 2$ (e.g., see d02nb). Set **maxstp** $= 0$ if no limit is to be imposed.

10: **mxhnil – int32 scalar**

The maximum number of warnings printed (if **itrace** ≥ 0, e.g., see d02nb) per problem when $t + h = t$ on a step ($h =$ current step size). If **mxhnil** ≤ 0, a default value of 10 is assumed.

11: **norm_p – string**

Indicates the type of norm to be used.

norm_p $=$ 'M'

Maximum norm.

norm_p $=$ 'A'

Averaged L2 norm.

norm_p $=$ 'D'

Is the same as 'A'.

If $vnorm$ denotes the norm of the vector v of length n, then for the averaged L2 norm

$$vnormB = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{v_i}{w_i}\right)^2},$$

while for the maximum norm

$$vnorm = \max_{1\leq i\leq n}\left|\frac{v_i}{w_i}\right|.$$

If you wish to weight the maximum norm or the L2 norm, then **rtol** and **atol** should be scaled appropriately on input to the integrator (see under **itol** in the specification of the integrator for the formulation of the weight vector w_i from **rtol** and **atol**, e.g., see d02nb).

Only the first character to the actual parameter **norm_p** is passed to d02mv; hence it is permissible for the actual argument to be more descriptive, e.g., 'Maximum', 'Average L2' or 'Default' in a call to d02mv.

Constraint: **norm_p** = 'M', 'A' or 'D'.

12: **rwork**$(\mathbf{50} + \mathbf{4} \times \mathbf{ldysav})$ **– double array**

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

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **con**$(\mathbf{3})$ **– double array**

The values actually to be used by the integration function.

2: **rwork**$(\mathbf{50} + \mathbf{4} \times \mathbf{ldysav})$ **– double array**

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

3: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

On entry, **ldysav** < 1,
or **sdysav** $<$ **maxord** $+ 3$,
or **maxord** < 0,
or **maxord** > 5,
or invalid value for element of the array **con**,
or **norm_p** $\neq$ 'M', 'A' or 'D'.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

```
neqmax = int32(6);
ny2dim = int32(8);
maxord = int32(0);
con = [0;
     0;
     0];
tcrit = 3.141592653589793;
hmin = 1e-10;
hmax = 0;
h0 = 0;
maxstp = int32(5000);
mxhnil = int32(5);
normtype = 'AVERAGE-L2';
rwork = zeros(74,1);
[conOut, rworkOut, ifail] = ...
      d02mv(neqmax, ny2dim, maxord, con, tcrit, hmin, hmax, h0, maxstp,
mxhnil, normtype, rwork)
```

```
conOut =
           2
          10
        1000
rworkOut =
     array elided
ifail =
           0
```