NAG Library Routine Document

D02PYF

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

D02PYF provides details about an integration performed by either D02PCF or D02PDF.

2 Specification

```
SUBROUTINE D02PYF (TOTFCN, STPCST, WASTE, STPSOK, HNEXT, IFAIL)

INTEGER            TOTFCN, STPCST, STPSOK, IFAIL
REAL (KIND=nag_wp) WASTE, HNEXT
```

3 Description

D02PYF and its associated routines (D02PCF, D02PDF, D02PVF, D02PWF, D02PXF and D02PZF) solve the initial value problem for a first-order system of ordinary differential equations. The routines, based on Runge–Kutta methods and derived from RKSUITE (see Brankin *et al.* (1991)), integrate

$$y' = f(t, y) \qquad \text{given} \qquad y(t_0) = y_0,$$

where y is the vector of n solution components and t is the independent variable.

After a call to D02PCF or D02PDF, D02PYF can be called to obtain information about the cost of the integration and the size of the next step.

4 References

Brankin R W, Gladwell I and Shampine L F (1991) RKSUITE: A suite of Runge–Kutta codes for the initial value problems for ODEs *SoftReport 91-S1* Southern Methodist University

5 Parameters

1: TOTFCN – INTEGER *Output*

On exit: the total number of evaluations of f used in the primary integration so far; this includes evaluations of f for the secondary integration specified by a prior call to D02PVF with ERRASS = .TRUE..

2: STPCST – INTEGER *Output*

On exit: the cost in terms of number of evaluations of f of a typical step with the method being used for the integration. The method is specified by the parameter METHOD in a prior call to D02PVF.

3: WASTE – REAL (KIND=nag_wp) *Output*

On exit: the number of attempted steps that failed to meet the local error requirement divided by the total number of steps attempted so far in the integration. A 'large' fraction indicates that the integrator is having trouble with the problem being solved. This can happen when the problem is 'stiff' and also when the solution has discontinuities in a low-order derivative.

4: STPSOK – INTEGER *Output*

On exit: the number of accepted steps.

5: HNEXT – REAL (KIND=nag_wp) *Output*

On exit: the step size the integrator will attempt to use for the next step.

6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

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, if you are 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.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (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 or warnings detected by the routine:

IFAIL $= 1$

An invalid call to D02PYF has been made, for example without a previous call to D02PCF or D02PDF. You cannot continue integrating the problem.

7 Accuracy

Not applicable.

8 Further Comments

When a secondary integration has taken place, that is when global error assessment has been specified using ERRASS = .TRUE. in a prior call to D02PVF, then the approximate extra number of evaluations of f used is given by $2 \times$ STPSOK $\times$ STPCST for METHOD $= 2$ or 3 and $3 \times$ STPSOK $\times$ STPCST for METHOD $= 1$.

9 Example

See Section 9 in D02PCF, D02PDF, D02PWF, D02PXF and D02PZF.