NAG Library Routine Document

D02PHF

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

D02PHF is a reverse communication routine that computes the interpolant for evaluation by D02PJF anywhere on an integration step taken by D02PGF. The direct communication version of the D02PHF and D02PJF pair is D02PSF. A significant difference in functionality between the forward and reverse communication versions is that D02PHF and D02PJF can interpolate for the high-order Runge–Kutta method.

2 Specification

```
SUBROUTINE D02PHF (IREVCM, N, NWANT, T, Y, YP, WCOMM, LWCOMM, IWSAV,       &
                   RWSAV, IFAIL)

INTEGER            IREVCM, N, NWANT, LWCOMM, IWSAV(130), IFAIL
REAL (KIND=nag_wp) T, Y(N), YP(N), WCOMM(LWCOMM), RWSAV(32*N+350)
```

3 Description

D02PHF and its associated routines (D02PGF, D02PJF, D02PQF, D02PRF, D02PTF and D02PUF) 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

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

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

D02PGF computes the solution at the end of an integration step. Using the information computed on that step D02PHF computes the interpolant which can be evaluated at any point on that step by D02PJF. If METHOD $= 1$ or -1 then there is enough information available from the stages of the last step to provide an interpolant of sufficient order of accuracy; no further derivative evaluations will therefore be requested. If METHOD $= 2$ or -2 then the interpolant is an order 8 continuous Runge–Kutta process that requires a further 3 stages of derivative evaluations that will be requested in turn before a final exit. If METHOD $= 3$ or -3 was specified in the call to setup routine D02PQF then the interpolant is a continuous Runge–Kutta process requiring a further 7 stages of derivative evaluations that will be requested in turn.

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 Arguments

Note: this routine uses **reverse communication.** Its use involves an initial entry, intermediate exits and re-entries, and a final exit, as indicated by the argument **IREVCM**. Between intermediate exits and re-entries, **all arguments other than those specified by the value of IREVCM must remain unchanged**.

1: IREVCM – INTEGER *Input/Output*

On initial entry: IREVCM must be set to zero to indicate that the interpolant for a new step is being taken.

On intermediate re-entry: IREVCM should remain unchanged.

On intermediate exit: IREVCM returns a value 1 to indicate that a function evaluation is required prior to re-entry; the value of the derivatives must be returned in YP where the value of t is supplied in T and the values $y(t)$ are supplied in the array Y.

On final exit:

IREVCM $= -1$
Successful exit; RWSAV and WCOMM contain details of the interpolant.

IREVCM $= -2$
Error exit; IFAIL should be interrogated to determine the nature of the error.

2: N – INTEGER *Input*

On entry: n, the number of ordinary differential equations in the system to be solved by the integration routine.

Constraint: N ≥ 1. This must be the same value as supplied in a previous call to D02PQF.

3: NWANT – INTEGER *Input*

On entry: the number of components of the solution to be computed. The first NWANT components are evaluated.

Constraint: $1 \leq$ NWANT $\leq$ N.

4: T – REAL (KIND=nag_wp) *Output*

On intermediate exit: T contains the value of the independent variable t at which the derivatives y' are to be evaluated.

On final exit: contains no useful information.

5: Y(N) – REAL (KIND=nag_wp) array *Output*

On intermediate exit: Y contains the value of the solution y at which the derivatives y' are to be evaluated.

On final exit: contains no useful information.

6: YP(N) – REAL (KIND=nag_wp) array *Input*

On initial entry: need not be set.

On intermediate re-entry: YP must contain the values of the derivatives y'_i for the given values of the arguments t, y_i.

7: WCOMM(LWCOMM) – REAL (KIND=nag_wp) array *Communication Array*

On entry: WCOMM need not be set.

On intermediate re-entry: WCOMM contains the partial computation of the polynomial coefficients corresponding to a continuous Runge–Kutta process for interpolating medium and high order Runge–Kutta methods.

On final exit: if METHOD $= 2, -2, 3$ or -3, WCOMM contains details of the interpolant which must be passed unchanged to D02PJF for evaluation of the interpolant.

8: LWCOMM – INTEGER *Input*

On entry: length of WCOMM.

If in a previous call to D02PQF:

METHOD $= 1$ or -1, LWCOMM must be at least 1.

METHOD $= 2$ or -2, LWCOMM must be at least $\mathrm{N} + \max(\mathrm{N}, 5 \times \mathrm{NWANT})$.

METHOD $= 3$ or -3, LWCOMM must be at least $8 \times \mathrm{NWANT}$.

9: IWSAV(130) – INTEGER array *Communication Array*

10: RWSAV($32 \times \mathrm{N} + 350$) – REAL (KIND=nag_wp) array *Communication Array*

On entry: these must be the same arrays supplied in a previous call D02PGF. They must remain unchanged between calls.

On exit: information about the integration for use on subsequent calls to D02PGF, D02PJF or other associated routines.

11: IFAIL – INTEGER *Input/Output*

On initial entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation 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, because for this routine the values of the output arguments may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On final 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 $= 1$

On entry, a previous call to the setup routine has not been made or the communication arrays have become corrupted, or a catastrophic error has already been detected elsewhere.
You cannot continue integrating the problem.

On entry, LWCOMM $= \langle value \rangle$, N $= \langle value \rangle$ and NWANT $= \langle value \rangle$.
Constraint: for METHOD $= -2$ or 2, LWCOMM $\geq \mathrm{N} + \max(\mathrm{N}, 5 \times \mathrm{NWANT})$.

On entry, LWCOMM $= \langle value \rangle$.
Constraint: for METHOD $= -1$ or 1, LWCOMM ≥ 1.

On entry, LWCOMM $= \langle value \rangle$ and NWANT $= \langle value \rangle$.
Constraint: for METHOD $= -3$ or 3, LWCOMM $\geq 8 \times \mathrm{NWANT}$.

On entry, N $= \langle value \rangle$, but the value passed to the setup routine was N $= \langle value \rangle$.

On entry, NWANT $= \langle value \rangle$ and N $= \langle value \rangle$.
Constraint: $1 \leq \mathrm{NWANT} \leq \mathrm{N}$.

You cannot call this routine after the integrator has returned an error.

You cannot call this routine after the range integrator has been called.

You cannot call this routine before you have called the step integrator.

IFAIL $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -399$

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

The computed values will be of a similar accuracy to that computed by D02PGF.

8 Parallelism and Performance

D02PHF 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 X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

None.

10 Example

See Section 10 in D02PGF.