NAG Library Routine Document

D02PJF

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

D02PJF evaluates the interpolant calculated by D02PHF, following an integration step performed by D02PGF to solve an initial value problem.

2 Specification

```
SUBROUTINE D02PJF (ICHECK, N, NWANT, T, IDERIV, SOL, WCOMM, LWCOMM,        &
                   IWSAV, RWSAV, IFAIL)

INTEGER           ICHECK, N, NWANT, IDERIV, LWCOMM, IWSAV(130), IFAIL
REAL (KIND=nag_wp) T, SOL(NWANT), WCOMM(LWCOMM), RWSAV(32*N+350)
```

3 Description

When integrating using the reverse communication Runge–Kutta integrator D02PGF, the solution or its derivatives can be obtained inexpensively between steps by interpolation. D02PHF is called after a step by D02PGF from a previous value of $t\ (=t_{k-1})$ to its current value, $t=t_k$ (i.e., a kth successful time-step has been taken). D02PJF can then be called to evaluate interpolated approximations of the function or its derivatives at any value of t in the interval (t_{k-1}, t_k).

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

1:   ICHECK – INTEGER   *Input*

    On entry: indicates whether consistency checks on input arguments should be performed

    ICHECK $\neq 1$
        Don't perform checks on input arguments.

    ICHECK $= 1$
        Perform consistency checks on input arguments.

    It is recommended to use ICHECK $= 1$ on the first call following a call to D02PHF and to set ICHECK $\neq 1$ on subsequent calls within the last step to avoid the overhead of argument checking.

2:   N – INTEGER   *Input*

    On entry: n, the dimension of the system of ODEs being integrated.

    Constraint: this must be the same value as supplied in a previous call to D02PQF.

3:   NWANT – INTEGER   *Input*

    On entry: only the first NWANT system components to be computed. This should be the same value as passed to D02PHF when computing the interpolant.

    Constraint: NWANT $=$ NWANT passed to D02PHF.

4:      T – REAL (KIND=nag_wp)   *Input*

On entry: t, the value of the independent variable where a solution is desired. Although any value of t can be supplied, accurate solutions can only be obtained for values in the range of the last time-step taken by D02PGF.

5:      IDERIV – INTEGER   *Input*

On entry:

IDERIV $= 0$
     Compute approximations to the first NWANT components of the solution $y(t)$.

IDERIV $= 1$
     Compute approximations to the first NWANT components of the first derivatives of the solution $y'(t)$.

Constraint: IDERIV $= 0$ or 1.

6:      SOL(NWANT) – REAL (KIND=nag_wp) array   *Output*

On exit:

IDERIV $= 0$
     The first NWANT components of the solution $y(t)$.

IDERIV $= 1$
     The first NWANT components of the first derivatives of the solution $y'(t)$.

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

On entry: this must be the same array supplied in a previous call to D02PHF. It must remain unchanged between calls.

8:      LWCOMM – INTEGER   *Input*

On entry: length of WCOMM. This should be the same value as supplied in a previous call to D02PHF.

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 $N + \max(N, 5 \times NWANT)$.

          METHOD $= 3$ or -3, LWCOMM $\geq 8 \times NWANT$.

9:      IWSAV(130) – INTEGER array   *Communication Array*
10:     RWSAV($32 \times N + 350$) – 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, D02PHF or other associated routines.

11:     IFAIL – INTEGER   *Input/Output*

On 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, if you are not familiar with this argument, 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

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, IDERIV = $\langle value \rangle$.
Constraint: IDERIV = 0 or 1.

On entry, LWCOMM = $\langle value \rangle$, N = $\langle value \rangle$ and NWANT = $\langle value \rangle$.
Constraint: for METHOD = −2 or 2, LWCOMM $\geq$ N + max(N, 5 × NWANT).

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

On entry, LWCOMM = $\langle value \rangle$ and NWANT = $\langle value \rangle$.
Constraint: for METHOD = −3 or 3, LWCOMM $\geq$ 8 × NWANT.

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

On entry, NWANT = $\langle value \rangle$, but on interpolation setup NWANT = $\langle value \rangle$.
Constraint: NWANT must be unchanged from setup.

The previous call to the interpolation setup routine returned an error.

You cannot call this routine before you have called the interpolation setup.

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

D02PJF is not threaded in any implementation.

9   Further Comments

None.

10 Example

See Section 10 in D02PGF.