NAG Fortran Library Routine Document

D03PZF

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

This routine interpolates in the spatial co-ordinate the solution and derivative of a system of partial differential equations (PDEs). The solution must first be computed using one of the finite difference scheme routines D03PCF/D03PCA, D03PHF/D03PHA or D03PPF/D03PPA, or one of the Keller box scheme routines D03PEF, D03PKF or D03PRF.

2 Specification

```
      SUBROUTINE D03PZF(NPDE, M, U, NPTS, X, XP, INTPTS, ITYPE, UP, IFAIL)
      INTEGER            NPDE, M, NPTS, INTPTS, ITYPE, IFAIL
      real               U(NPDE,NPTS), X(NPTS), XP(INTPTS),
     1                   UP(NPDE,INTPTS,ITYPE)
```

3 Description

D03PZF is an interpolation routine for evaluating the solution of a system of partial differential equations (PDEs), at a set of user-specified points. The solution of the system of equations (possibly with coupled ordinary differential equations) must be computed using a finite difference scheme routine or a Keller box scheme routine on a set of mesh points. D03PZF can then be employed to compute the solution at a set of points anywhere in the range of the mesh. It can also evaluate the first spatial derivative of the solution. The routine uses linear interpolation for approximating the solution.

4 References

None.

5 Parameters

Note: the parameters X, M, U, NPTS and NPDE must be supplied unchanged from the PDE routine.

1: NPDE – INTEGER *Input*

On entry: the number of PDEs.

Constraint: NPDE $\geq$ 1.

2: M – INTEGER *Input*

On entry: the co-ordinate system used. If the call to D03PZF follows one of the finite difference routines then M must be the same parameter M as used by the finite difference routines. For the Keller box scheme routines only Cartesian co-ordinate systems are valid and so M **must** be set to zero. No check will be made by D03PZF in this case.

M = 0

indicates Cartesian co-ordinates

M = 1

indicates cylindrical polar co-ordinates

M = 2

indicates spherical polar co-ordinates

Constraints:

$0 \leq \mathrm{M} \leq 2$ following a finite difference routine.
$\mathrm{M} = 0$ following a Keller box scheme routine.

3: U(NPDE,NPTS) – ***real*** array *Input*

On entry: the PDE part of the original solution returned in the parameter U by the PDE routine.

Constraint: NPDE ≥ 1.

4: NPTS – INTEGER *Input*

On entry: the number of mesh points.

Constraint: NPTS ≥ 3.

5: X(NPTS) – ***real*** array *Input*

On entry: X(i), for $i = 1, 2, \ldots,$ NPTS, must contain the mesh points as used by the PDE routine.

6: XP(INTPTS) – ***real*** array *Input*

On entry: XP(i), for $i = 1, 2, \ldots,$ INTPTS, must contain the spatial interpolation points.

Constraint: X$(1) \leq$ XP$(1) <$ XP$(2) < \ldots <$ XP(INTPTS) $\leq$ X(NPTS).

7: INTPTS – INTEGER *Input*

On entry: the number of interpolation points.

Constraint: INTPTS ≥ 1.

8: ITYPE – INTEGER *Input*

On entry: specifies the interpolation to be performed.

If ITYPE $= 1$, the solutions at the interpolation points are computed. If ITYPE $= 2$, both the solutions and their first derivatives at the interpolation points are computed.

Constraint: ITYPE $= 1$ or 2.

9: UP(NPDE,INTPTS,ITYPE) – ***real*** array *Output*

On exit: if ITYPE $= 1$, UP$(i, j, 1)$, contains the value of the solution $U_i(x_j, t_{\mathrm{out}})$, at the interpolation points $x_j =$ XP(j), for $j = 1, 2, \ldots,$ INTPTS; $i = 1, 2, \ldots,$ NPDE.

If ITYPE $= 2$, UP$(i, j, 1)$ contains $U_i(x_j, t_{\mathrm{out}})$ and UP$(i, j, 2)$ contains $\frac{\partial U_i}{\partial x}$ at these points.

10: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

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

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, for users 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.**

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, ITYPE $\neq 1$ or 2,
or INTPTS < 1,
or NPDE < 1,
or NPTS < 3,
or M $\neq 0$, 1 or 2,
or the mesh points X(i), for $i = 1, 2, \ldots,$ NPTS, are not in strictly increasing order.

IFAIL = 2

On entry, the interpolation points XP(i), for $i = 1, 2, \ldots,$ INTPTS, are not in strictly increasing order.

IFAIL = 3

The user is attempting extrapolation, that is, one of the interpolation points XP(i), for some i, lies outside the interval [X(1),X(NPTS)]. Extrapolation is not permitted.

7 Accuracy

See the PDE routine documents.

8 Further Comments

None.

9 Example

See Section 9 of the documents for D03PCF/D03PCA, D03PPF/D03PPA and D03PRF.
