NAG C Library Chapter Introduction

d03 – Partial Differential Equations

Contents

1 Scope of the Chapter

This chapter is concerned with the numerical solution of partial differential equations. Currently only solvers for parabolic and hyperbolic equations are included.

2 Background to the Problems

The definition of a partial differential equation problem includes not only the equation itself but also the domain of interest and appropriate subsidiary conditions. Indeed, partial differential equations are usually classified as elliptic, hyperbolic or parabolic according to the form of the equation **and** the form of the subsidiary conditions which must be assigned to produce a well-posed problem. The functions in this chapter will often call upon functions from other chapters, such as Chapter f04 (Simultaneous Linear Equations) and Chapter d02 (Ordinary Differential Equations). Other chapters also contain relevant functions, in particular Chapter d06 (Mesh Generation) and Chapter f11 (Sparse Linear Algebra).

The classification of partial differential equations is easily described in the case of **linear** equations of the **second order** in two independent variables, i.e., equations of the form

$$au_{xx} + 2bu_{xy} + cu_{yy} + du_x + eu_y + fu + g = 0, \tag{1}$$

where a, b, c, d, e, f and g are functions of x and y only. Equation (1) is called elliptic, hyperbolic or parabolic according to whether $ac - b^2$ is positive, negative or zero, respectively. Useful definitions of the concepts of elliptic, hyperbolic and parabolic character can also be given for differential equations in more than two independent variables, for systems and for nonlinear differential equations.

For **elliptic** equations, of which Laplace's equation

$$u_{xx} + u_{yy} = 0 \tag{2}$$

is the simplest example of second order, the subsidiary conditions take the form of **boundary** conditions, i.e., conditions which provide information about the solution at all points of a **closed** boundary. For example, if equation (2) holds in a plane domain D bounded by a contour C, a solution u may be sought subject to the condition

$$u = f \quad \text{on} \quad C, \tag{3}$$

where f is a given function. The condition (3) is known as a Dirichlet boundary condition. Equally common is the Neumann boundary condition

$$u' = g \quad \text{on} \quad C, \tag{4}$$

which is one form of a more general condition

$$u' + fu = g \quad \text{on} \quad C, \tag{5}$$

where u' denotes the derivative of u normal to the contour C, and f and g are given functions. Provided that f and g satisfy certain restrictions, condition (5) yields a well-posed **boundary value problem** for Laplace's equation. In the case of the Neumann problem, one further piece of information, e.g., the value of u at a particular point, is necessary for uniqueness of the solution. Boundary conditions similar to the above are applicable to more general second-order elliptic equations, whilst two such conditions are required for equations of fourth order.

For **hyperbolic** equations, the wave equation

$$u_{tt} - u_{xx} = 0 \tag{6}$$

is the simplest example of second order. It is equivalent to a first-order system

$$u_t - v_x = 0, \quad v_t - u_x = 0. \tag{7}$$

The subsidiary conditions may take the form of **initial** conditions, i.e., conditions which provide information about the solution at points on a suitable **open** boundary. For example, if equation (6) is satisfied for $t > 0$, a solution u may be sought such that

$$u(x,0) = f(x), \quad u_t(x,0) = g(x), \tag{8}$$

where f and g are given functions. This is an example of an **initial value problem**, sometimes known as Cauchy's problem.

For **parabolic** equations, of which the heat conduction equation

$$u_t - u_{xx} = 0 \tag{9}$$

is the simplest example, the subsidiary conditions always include some of **initial** type and may also include some of **boundary** type. For example, if equation (9) is satisfied for $t > 0$ and $0 < x < 1$, a solution u may be sought such that

$$u(x, 0) = f(x), \quad 0 < x < 1, \tag{10}$$

and

$$u(0, t) = 0, \quad u(1, t) = 1, \quad t > 0. \tag{11}$$

This is an example of a mixed **initial/boundary value problem**.

For all types of partial differential equations, finite difference methods (Mitchell and Griffiths (1980)) and finite element methods (Wait and Mitchell (1985)) are the most common means of solution and such methods obviously feature prominently either in this chapter or in the companion NAG Finite Element Library. Some of the utility functions in this chapter are concerned with the solution of the large sparse systems of equations which arise from finite difference and finite element methods. Further functions for this purpose are provided in Chapter f11.

Alternative methods of solution are often suitable for special classes of problems. For example, the method of characteristics is the most common for hyperbolic equations involving time and one space dimension (Smith (1985)). The method of lines (see Mikhlin and Smolitsky (1967)) may be used to reduce a parabolic equation to a (stiff) system of ordinary differential equations, which may be solved by means of functions from Chapter d02 (Ordinary Differential Equations). Similarly, integral equation or boundary element methods (Jaswon and Symm (1977)) are frequently used for elliptic equations. Typically, in the latter case, the solution of a boundary value problem is represented in terms of certain boundary functions by an integral expression which satisfies the differential equation throughout the relevant domain. The boundary functions are obtained by applying the given boundary conditions to this representation. Implementation of this method necessitates discretization of only the boundary of the domain, the dimensionality of the problem thus being effectively reduced by one. The boundary conditions yield a full system of simultaneous equations, as opposed to the sparse systems yielded by finite difference and finite element methods, but the full system is usually of much lower order. Solution of this system yields the boundary functions, from which the solution of the problem may be obtained, by quadrature, as and where required.

3 Recommendations on Choice and Use of Available Functions

3.1 Hyperbolic Equations

See Section 3.5.

3.2 Parabolic Equations

There are five functions available for solving general parabolic equations in one space dimension:

nag_pde_parab_1d_fd (d03pcc),

nag_pde_parab_1d_coll (d03pdc),

nag_pde_parab_1d_fd_ode (d03phc),

nag_pde_parab_1d_coll_ode (d03pjc),

nag_pde_parab_1d_fd_ode_remesh (d03ppc).

Equations may include nonlinear terms but the true derivative u_t should occur linearly and equations should usually contain a second-order space derivative u_{xx}. There are certain restrictions on the coefficients to try to ensure that the problems posed can be solved by the above functions.

The method of solution is to discretize the space derivatives using finite differences or collocation, and to solve the resulting system of ordinary differential equations using a ‘stiff’ solver.

nag_pde_parab_1d_fd (d03pcc) and nag_pde_parab_1d_coll (d03pdc) can solve a system of parabolic equations of the form

$$\sum_{j=1}^{n} P_{ij}(x,t,U,U_x)\frac{\partial U_j}{\partial t} + Q_i(x,t,U,U_x) = x^{-m}\frac{\partial}{\partial x}(x^m R_i(x,t,U,U_x)),$$

where $i = 1,2,\ldots,n,\ a \le x \le b,\ t \ge t_0$.

The parameter m allows the function to handle different coordinate systems easily (Cartesian, cylindrical polars and spherical polars). nag_pde_parab_1d_fd (d03pcc) uses a finite differences spatial discretization and nag_pde_parab_1d_coll (d03pdc) uses a collocation spatial discretization.

nag_pde_parab_1d_fd_ode (d03phc) and nag_pde_parab_1d_coll_ode (d03pjc) are similar to nag_pde_parab_1d_fd (d03pcc) and nag_pde_parab_1d_coll (d03pdc) respectively, except that they provide scope for coupled differential-algebraic systems. This extended functionality allows for the solution of more complex and more general problems, e.g., periodic boundary conditions and integro-differential equations.

nag_pde_parab_1d_fd_ode_remesh (d03ppc) is similar to nag_pde_parab_1d_fd_ode (d03phc) but allows remeshing to take place in the spatial direction. This facility can be very useful when the nature of the solution in the spatial direction varies considerably over time.

3.3 Black–Scholes Equations

nag_pde_bs_1d (d03ncc) solves the Black–Scholes equation

$$\frac{\partial f}{\partial t} + (r-q)S\frac{\partial f}{\partial S} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 f}{\partial S^2} = rf$$

$$S_{\text{min}} < S < S_{\text{max}}, \quad t_{\text{min}} < t < t_{\text{max}},$$

for the value f of a European or American, put or call stock option. The parameters r, q and σ may each be either constant or time-dependent. The values of the Greeks are also returned.

In certain cases an analytic solution of the Black–Scholes equation is available. In these cases the solution may be computed by nag_pde_bs_1d_analytic (d03ndc).

3.4 First-order Systems in One Space Dimension

There are three functions available for solving systems of first-order partial differential equations:

nag_pde_parab_1d_keller (d03pec),

nag_pde_parab_1d_keller_ode (d03pkc),

nag_pde_parab_1d_keller_ode_remesh (d03prc).

Equations may include nonlinear terms but the time derivative should occur linearly. There are certain restrictions on the coefficients to ensure that the problems posed can be solved by the above functions.

The method of solution is to discretize the space derivatives using the Keller box scheme and to solve the resulting system of ordinary differential equations using a ‘stiff’ solver.

nag_pde_parab_1d_keller (d03pec) is designed to solve a system of the form

$$\sum_{j=1}^{n} P_{ij}(x,t,U,U_x)\frac{\partial U_j}{\partial t} + Q_i(x,t,U,U_x) = 0,$$

where $i = 1,2,\ldots,n,\ a \le x \le b,\ t \ge t_0$.

nag_pde_parab_1d_keller_ode (d03pkc) is similar to nag_pde_parab_1d_keller (d03pec) except that it provides scope for coupled differential algebraic systems. This extended functionality allows for the solution of more complex problems.

nag_pde_parab_1d_keller_ode_remesh (d03prc) is similar to nag_pde_parab_1d_keller_ode (d03pkc) but allows remeshing to take place in the spatial direction. This facility can be very useful when the nature of the solution in the spatial direction varies considerably over time.

nag_pde_parab_1d_keller (d03pec), nag_pde_parab_1d_keller_ode (d03pkc) or nag_pde_parab_1d_keller_ode_remesh (d03prc) may also be used to solve systems of higher or mixed order partial differential equations which have been reduced to first-order. Note that in general these functions are unsuitable for hyperbolic first-order equations, for which an appropriate upwind discretization scheme should be used (see Section 3.5 for example).

3.5 Convection-diffusion Systems

There are three functions available for solving systems of convection-diffusion equations with optional source terms:

nag_pde_parab_1d_cd (d03pfc),

nag_pde_parab_1d_cd_ode (d03plc),

nag_pde_parab_1d_cd_ode_remesh (d03psc).

Equations may include nonlinear terms but the time derivative should occur linearly. There are certain restrictions on the coefficients to ensure that the problems posed can be solved by the above functions, in particular the system must be posed in conservative form (see below). The functions may also be used to solve hyperbolic convection-only systems.

Convection terms are discretized using an upwind scheme involving a numerical flux function based on the solution of a Riemann problem at each mesh point (LeVeque (1990)); and diffusion and source terms are discretized using central differences. The resulting system of ordinary differential equations is solved using a 'stiff' solver. In the case of Euler equations for a perfect gas various approximate and exact Riemann solvers are provided in nag_pde_parab_1d_euler_roe (d03puc), nag_pde_parab_1d_euler_osher (d03pvc), nag_pde_parab_1d_euler_hll (d03pwc) and nag_pde_parab_1d_euler_exact (d03pxc). These functions may be used in conjunction with nag_pde_parab_1d_cd (d03pfc), nag_pde_parab_1d_cd_ode (d03plc) and nag_pde_parab_1d_cd_ode_remesh (d03psc).

nag_pde_parab_1d_cd (d03pfc) is designed to solve systems of the form

$$\sum_{j=1}^{n} P_{ij}(x,t,U)\frac{\partial U_j}{\partial t} + \frac{\partial}{\partial x}F_i(x,t,U) = C_i(x,t,U)\frac{\partial}{\partial x}D_i(x,t,U,U_x) + S_i(x,t,U),$$

or hyperbolic convection-only systems of the form

$$\sum_{j=1}^{n} P_{ij}(x,t,U)\frac{\partial U_j}{\partial t} + \frac{\partial F_i(x,t,U)}{\partial x} = 0,$$

where $i = 1, 2, \ldots, n,\ a \leq x \leq b,\ t \geq t_0$.

nag_pde_parab_1d_cd_ode (d03plc) is similar to nag_pde_parab_1d_cd (d03pfc) except that it provides scope for coupled differential algebraic systems. This extended functionality allows for the solution of more complex problems.

nag_pde_parab_1d_cd_ode_remesh (d03psc) is similar to nag_pde_parab_1d_cd_ode (d03plc) but allows remeshing to take place in the spatial direction. This facility can be very useful when the nature of the solution in the spatial direction varies considerably over time.

3.6 Automatic Mesh Generation

A range of mesh generation functions are available in Chapter d06.

3.7 Utility Functions

Functions are available in the Linear Algebra Chapters for the direct and iterative solution of linear equations. Here we point to some of the functions that may be of use in solving the linear systems that arise from finite difference or finite element approximations to partial differential equation solutions.

Chapters f11 should be consulted for further information and for the appropriate function documents. Decision trees for the solution of linear systems are given in Section 4 of the f04 Chapter Introduction.

The following functions allow the direct solution of symmetric positive-definite systems:

Band	nag_dpbtrf (f07hdc) and nag_dpbtrs (f07hec)
Variable band (skyline)	nag_real_cholesky_skyline (f01mcc) and nag_real_cholesky_skyline_solve (f04mcc)

and the following functions allow the iterative solution of symmetric positive-definite and symmetric-indefinite systems:

Sparse	nag_sparse_sym_chol_fac (f11jac), nag_sparse_sym_chol_sol (f11jcc) and nag_sparse_sym_sol (f11jec)

The latter two functions above are black box functions which include Incomplete Cholesky, SSOR or Jacobi preconditioning.

The following functions allow the direct solution of nonsymmetric systems:

Band	nag_dgbtrf (f07bdc) and nag_dgbtrs (f07bec)

and the following functions allow the iterative solution of nonsymmetric systems:

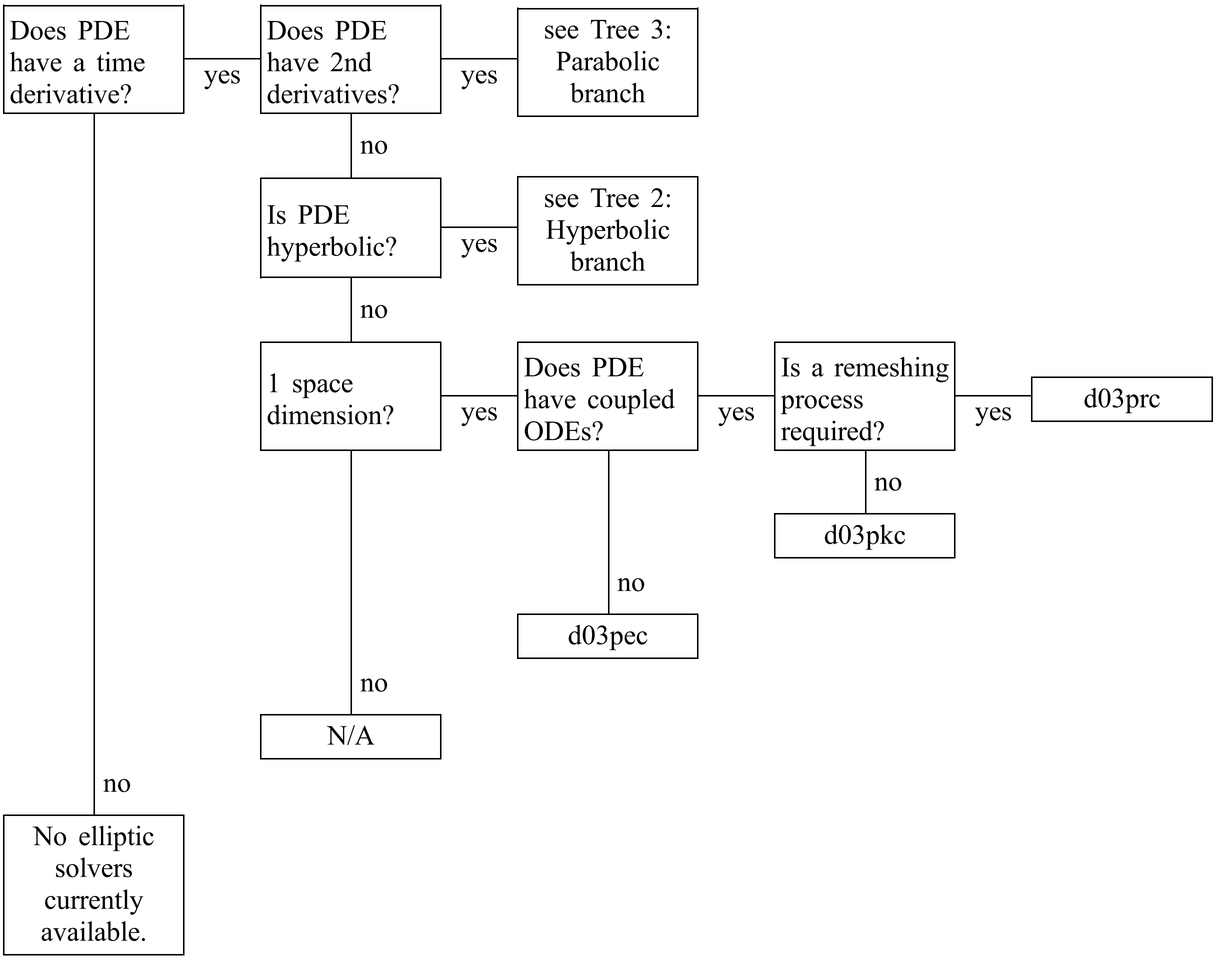
Sparse	nag_sparse_nsym_fac (f11dac), nag_sparse_nsym_fac_sol (f11dcc) and nag_sparse_nsym_sol (f11dec)

The latter two functions above are black box functions which include incomplete LU, SSOR and Jacobi preconditioning.

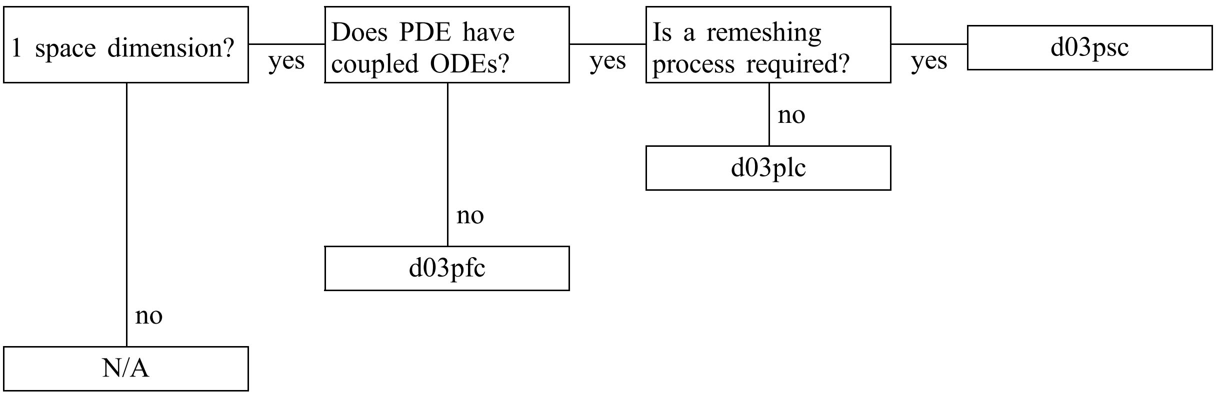
The functions nag_pde_interp_1d_fd (d03pzc) and nag_pde_interp_1d_coll (d03pyc) use linear interpolation to compute the solution to a parabolic problem and its first derivative at the user-specified points. nag_pde_interp_1d_fd (d03pzc) may be used in conjunction with nag_pde_parab_1d_fd (d03pcc), nag_pde_parab_1d_keller (d03pec), nag_pde_parab_1d_fd_ode (d03phc), nag_pde_parab_1d_keller_ode (d03pkc), nag_pde_parab_1d_fd_ode_remesh (d03ppc) and nag_pde_parab_1d_keller_ode_remesh (d03prc). nag_pde_interp_1d_coll (d03pyc) may be used in conjunction with nag_pde_parab_1d_coll (d03pdc) and nag_pde_parab_1d_coll_ode (d03pjc).

4 Decision Trees

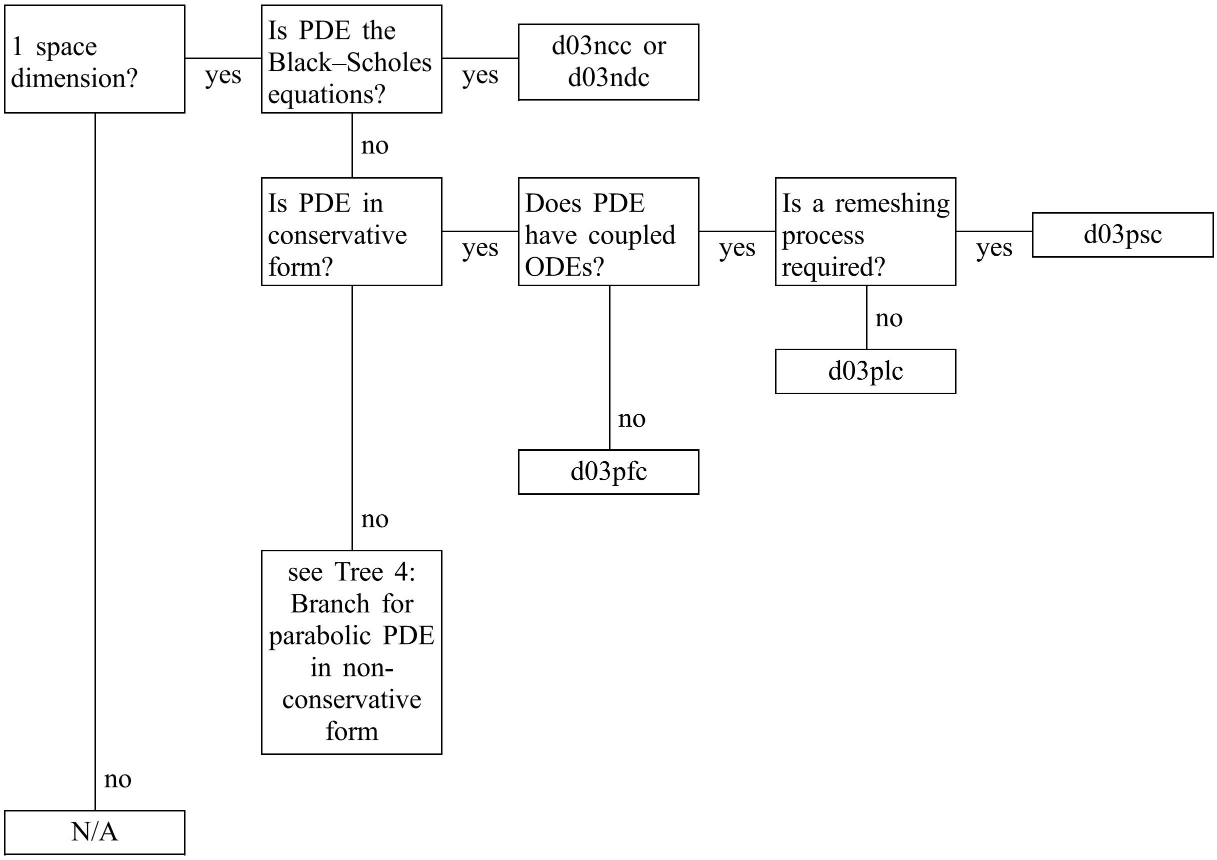
Tree 1



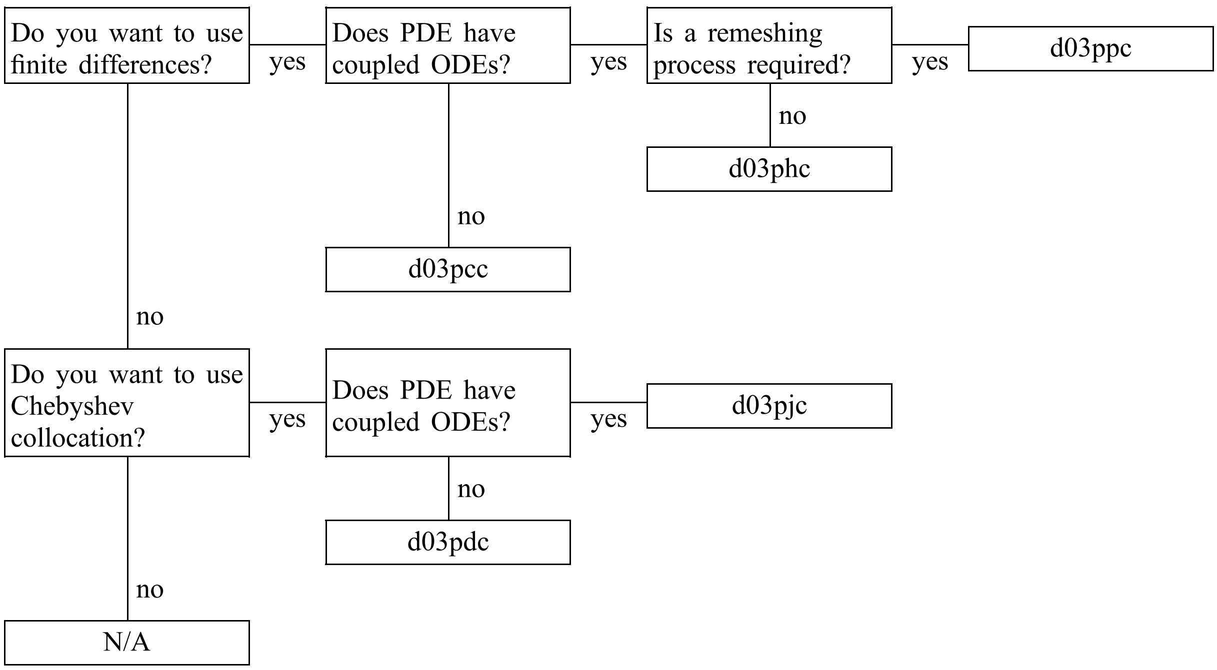
Tree 2: Hyperbolic branch



Tree 3: Parabolic branch



Tree 4: Branch for parabolic PDE in non-conservative form



5 Index

6 Functions Withdrawn or Scheduled for Withdrawal

None.

7 References

Ames W F (1977) *Nonlinear Partial Differential Equations in Engineering* (2nd Edition) Academic Press

Berzins M (1990) Developments in the NAG Library software for parabolic equations *Scientific Software Systems* (ed J C Mason and M G Cox) 59–72 Chapman and Hall

Jaswon M A and Symm G T (1977) *Integral Equation Methods in Potential Theory and Elastostatics* Academic Press

LeVeque R J (1990) *Numerical Methods for Conservation Laws* Birkhäuser Verlag

Mikhlin S G and Smolitsky K L (1967) *Approximate Methods for the Solution of Differential and Integral Equations* Elsevier

Mitchell A R and Griffiths D F (1980) *The Finite Difference Method in Partial Differential Equations* Wiley

Pennington S V and Berzins M (1994) New NAG Library software for first-order partial differential equations *ACM Trans. Math. Softw.* **20** 63–99

Richtmyer R D and Morton K W (1967) *Difference Methods for Initial-value Problems* (2nd Edition) Interscience

Smith G D (1985) *Numerical Solution of Partial Differential Equations: Finite Difference Methods* (3rd Edition) Oxford University Press

Swarztrauber P N and Sweet R A (1979) Efficient Fortran subprograms for the solution of separable elliptic partial differential equations *ACM Trans. Math. Software* **5** 352–364

Wait R and Mitchell A R (1985) *Finite Element Analysis and Application* Wiley