1
Introduction

1.1 Boundary Value Problems of Differential Equations

We discuss *numerical* solutions of problems involving ordinary differential equations (ODEs) or partial differential equations (PDEs), especially linear first- and second-order ODEs and PDEs, and problems involving systems of first-order differential equations.

A *differential equation* involves derivatives of an unknown function of one independent variable (say $u(x)$), or the partial derivatives of an unknown function of more than one independent variable (say $u(x, y)$, or $u(t, x)$, or $u(t, x, y, z)$, *etc.*). Differential equations have been used extensively to model many problems in daily life, such as pendulums, Newton's law of cooling, resistor and inductor circuits, population growth or decay, fluid and solid mechanics, biology, material sciences, economics, ecology, kinetics, thermodynamics, sports and computer sciences.[1] Examples include the Laplace equation for potentials, the Navier–Stokes equations in fluid dynamics, biharmonic equations for stresses in solid mechanics, and Maxwell equations in electromagnetics. For more examples and for the mathematical theory of PDEs, we refer the reader to Evans (1998) and references therein.

However, although differential equations have such wide applications, too few can be solved exactly in terms of elementary functions such as polynomials, $\log x$, e^x, trigonometric functions ($\sin x$, $\cos x$, ...), *etc.* and their combinations. Even if a differential equation can be solved analytically, considerable effort and sound mathematical theory are often needed, and the closed form of the solution may even turn out to be too messy to be useful. If the analytic solution of the differential equation is unavailable or too difficult to obtain, or

[1] There are other models in practice, for example, statistical models.

https://doi.org/10.1017/9781316678725.002 Published online by Cambridge University Press

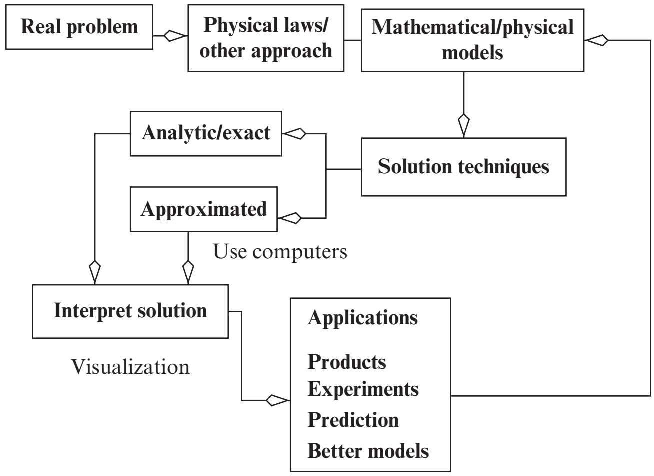

Figure 1.1. A flowchart of a problem-solving process.

takes some complicated form that is unhelpful to use, we may try to find an approximate solution. There are two traditional approaches:

1. Semi-analytic methods. Sometimes we can use series, integral equations, perturbation techniques, or asymptotic methods to obtain an approximate solution expressed in terms of simpler functions.
2. Numerical solutions. Discrete numerical values may represent the solution to a certain accuracy. Nowadays, these number arrays (and associated tables or plots) are obtained using computers, to provide effective solutions of many problems that were impossible to obtain before.

In this book, we mainly adopt the second approach and focus on numerical solutions using computers, especially the use of finite difference (FD) or finite element (FE) methods for differential equations. In Figure 1.1, we show a flowchart of the problem-solving process.

Some examples of ODE/PDEs are as follows.

1. Initial value problems (IVP). A canonical first-order system is

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(t_0) = \mathbf{y}_0\,; \tag{1.1}$$

and a single higher-order differential equation may be rewritten as a first-order system. For example, a second-order ODE

$$\begin{aligned} &u''(t) + a(t)u'(t) + b(t)u(t) = f(t), \\ &u(0) = u_0, \quad \boxed{u'(0) = v_0}. \end{aligned} \tag{1.2}$$

https://doi.org/10.1017/9781316678725.002 Published online by Cambridge University Press

can be converted into a first-order system by setting $y_1(t) = u$ and $y_2(t) = u'(t)$.

An ODE IVP can often be solved using Runge–Kutta methods, with adaptive time steps. In Matlab, there is the ODE-Suite which includes ode45, ode23, ode23s, ode15s, *etc.* For a stiff ODE system, either ode23s or ode15s is recommended; see Appendix for more details.

2. Boundary value problems (BVP). An example of an ODE BVP is

$$\begin{aligned} &u''(x) + a(x)u'(x) + b(x)u(x) = f(x), \quad 0 < x < 1, \\ &u(0) = u_0, \quad \boxed{u(1) = u_1}; \end{aligned} \tag{1.3}$$

and a PDE BVP example is

$$\begin{aligned} u_{xx} + u_{yy} &= f(x,y), \quad (x,y) \in \Omega, \\ u(x,y) &= u_0(x,y), \quad (x,y) \in \partial\Omega, \end{aligned} \tag{1.4}$$

where $u_{xx} = \frac{\partial^2 u}{\partial x^2}$ and $u_{yy} = \frac{\partial^2 u}{\partial y^2}$, in a domain Ω with boundary $\partial\Omega$. The above PDE is linear and classified as *elliptic*, and there are two other classifications for linear PDE, namely, *parabolic* and *hyperbolic*, as briefly discussed below.

3. BVP and IVP, *e.g.*,

$$\begin{aligned} &u_t = au_{xx} + f(x,t), \\ &u(0,t) = g_1(t), \quad u(1,t) = g_2(t), \qquad \text{BC} \\ &u(x,0) = u_0(x), \qquad \text{IC}, \end{aligned} \tag{1.5}$$

where BC and IC stand for boundary condition(s) and initial condition, respectively, where $u_t = \frac{\partial u}{\partial t}$.

4. Eigenvalue problems, *e.g.*,

$$\begin{aligned} &u''(x) = \lambda u(x), \\ &u(0) = 0, \quad u(1) = 0. \end{aligned} \tag{1.6}$$

In this example, both the function $u(x)$ (the *eigenfunction*) and the scalar λ (the *eigenvalue*) are unknowns.

5. Diffusion and reaction equations, *e.g.*,

$$\frac{\partial u}{\partial t} = \nabla \cdot (\beta \nabla u) + \mathbf{a} \cdot \nabla u + f(u) \tag{1.7}$$

where $\mathbf{a}$ is a vector, $\nabla \cdot (\beta \nabla u)$ is a diffusion term, $\mathbf{a} \cdot \nabla u$ is called an advection term, and $f(u)$ a reaction term.

https://doi.org/10.1017/9781316678725.002 Published online by Cambridge University Press

6. Systems of PDE. The incompressible Navier–Stokes model is an important nonlinear example:

$$\begin{aligned} &\rho\,(\mathbf{u}_t + (\mathbf{u}\cdot\nabla)\mathbf{u}) = \nabla p + \mu\Delta\mathbf{u} + \mathbf{F}, \\ &\nabla\cdot\mathbf{u} = 0. \end{aligned} \tag{1.8}$$

In this book, we will consider BVPs of differential equations in one dimension (1D) or two dimensions (2D). A linear second-order PDE has the following general form:

$$\begin{aligned} &a(x,y)u_{xx} + 2b(x,y)u_{xy} + c(x,y)u_{yy} \\ &\qquad + d(x,y)u_x + e(x,y)u_y + g(x,y)u(x,y) = f(x,y) \end{aligned} \tag{1.9}$$

where the coefficients are independent of $u(x,y)$ so the equation is linear in u and its partial derivatives. The solution of the 2D linear PDE is sought in some bounded domain Ω; and the classification of the PDE form (1.9) is:

- Elliptic if $b^2 - ac < 0$ for all $(x,y) \in \Omega$,
- Parabolic if $b^2 - ac = 0$ for all $(x,y) \in \Omega$, and
- Hyperbolic if $b^2 - ac > 0$ for all $(x,y) \in \Omega$.

The appropriate solution method typically depends on the equation class. For the first-order system

$$\frac{\partial\mathbf{u}}{\partial t} = A(\mathbf{x})\frac{\partial\mathbf{u}}{\partial\mathbf{x}}, \tag{1.10}$$

the classification is determined from the eigenvalues of the coefficient matrix $A(\mathbf{x})$.

Finite difference and finite element methods are suitable techniques to solve differential equations (ODEs and PDEs) numerically. There are other methods as well, for example, finite volume methods, collocation methods, spectral methods, *etc.*

1.1.1 Some Features of Finite Difference and Finite Element Methods

Many problems can be solved numerically by some finite difference or finite element methods. We strongly believe that any numerical analyst should be familiar with both methods and some important features listed below.

https://doi.org/10.1017/9781316678725.002 Published online by Cambridge University Press

Finite difference methods:

- Often relatively simple to use, and quite easy to understand.
- Easy to implement for regular domains, *e.g.*, rectangular domains in Cartesian coordinates, and circular or annular domains in polar coordinates.
- Their discretization and approximate solutions are pointwise, and the fundamental mathematical tool is the Taylor expansion.
- There are many fast solvers and packages for regular domains, *e.g.*, the Poisson solvers Fishpack (Adams et al.) and Clawpack (LeVeque, 1998).
- Difficult to implement for complicated geometries.
- Have strong regularity requirements (the existence of high-order derivatives).

Finite element methods:

- Very successful for structural (elliptic type) problems.
- Suitable approach for problems with complicated boundaries.
- Sound theoretical foundation, at least for elliptic PDE, using Sobolev space theory.
- *Weaker* regularity requirements.
- Many commercial packages, *e.g.*, Ansys, Matlab PDE Tool-Box, Triangle, and PLTMG.
- Usually coupled with multigrid solvers.
- Mesh generation can be difficult, but there are now many packages that do this, *e.g.*, Matlab, Triangle, Pltmg, Fidap, Gmsh, and Ansys.

1.2 Further Reading

This textbook provides an introduction to finite difference and finite element methods. There are many other books for readers who wish to become expert in finite difference and finite element methods.

For FD methods, we recommend Iserles (2008); LeVeque (2007); Morton and Mayers (1995); Strikwerda (1989) and Thomas (1995). The textbooks by Strikwerda (1989) and Thomas (1995) are classical, while Iserles (2008); LeVeque (2007) and Morton and Mayers (1995) are relatively new. With LeVeque (2007), the readers can find the accompanying Matlab code from the author's website.

A classic book on FE methods is Ciarlet (2002), while Johnson (1987) and Strang and Fix (1973) have been widely used as graduate textbooks. The series by Carey and Oden (1983) not only presents the mathematical background of FE methods, but also gives some details on FE method programming in Fortran. Newer textbooks include Braess (2007) and Brenner and Scott (2002).