Chapter 1
Differential equations, numerical methods and algebraic analysis

1.1 Introduction

Differential equations and numerical methods

Ordinary differential equations are at the heart of mathematical modelling. Typically ordinary differential equation systems arise as initial value problems

$$y'(x) = f\big(x, y(x)\big), \qquad y(x_0) = y_0 \in \mathbb{R}^N.$$

or, if y' does not depend directly on x,

$$y'(x) = f\big(y(x)\big), \qquad y(x_0) = y_0 \in \mathbb{R}^N. \tag{1.1 a}$$

The purpose of an equation like this is to describe the behaviour of a physical or other system and, at the same time, to predict future values of the time-dependant variable $y(x)$, whose components represent quantities of scientific interest.

It is often more convenient, in specific situations, to formulate (1.1 a) in different styles. For example, the components of $y(x)$ might represent differently named variables, and the formulation should express this. In other situations the problem being modelled might be more naturally represented using a system of second, or higher, order differential equations. However, we will usually use (1.1 a) as a standard form for a differential system.

Given $x > x_0$, the flow of (1.1 a) is the solution to this initial value problem evaluated at x. This is sometimes written as $\mathbf{e}^{(x-x_0)f}y_0$, but our preference will be to write it as $\mathit{flow}_{x-x_0}y_0$, where the nature of f is taken for granted.

The predictive power of differential equations is used throughout science, even when solutions cannot be obtained analytically, and this underlines the need for numerical methods. This usually means that we need to approximate $\mathit{flow}_h y_0$ to obtain a usable value of $y(x_0+h)$. This can be repeated computationally to obtain, in turn, $y(x_0+h)$, $y(x_0+2h)$, $y(x_0+3h)$, ...

Although many methods for carrying out the approximation to the flow are known, we will emphasize Runge–Kutta methods, because these consist of approximating

J. C. Butcher, *B-Series*, Springer Series in Computational Mathematics 55,
https://doi.org/10.1007/978-3-030-70956-3_1

the solution at $x_0 + nh$, $n = 1, 2, 3, \ldots$, step by step. As an example of these methods, choose one of the famous methods of Runge [82] (Runge, 1895), where the mapping $\mathcal{R}_h y_0 = y_1$ is defined by

$$y_1 = y_0 + hf\big(y_0 + \tfrac{1}{2}hf(y_0)\big). \tag{1.1 b}$$

Accuracy of numerical approximations

Accuracy of numerical methods will be approached, in this volume, through a study of the formal Taylor expansions of the solution, and of numerical approximations to the solution. The flavour of the questions that arise is both combinatorial and algebraic, because of the common structure of many of the formal expansions.

For the problem (1.1 a), we will need to compare the mappings flow_h and, for a particular Runge–Kutta method, the mapping $\mathcal{RK}_h$. This leads us to consider the difference

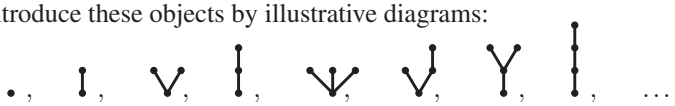
$$\mathit{flow}_h y_0 - \mathcal{RK}_h y_0.$$

If it were possible to expand this expression in a Taylor series, then it would be possible to seek methods for which the terms are zero up to some required power of h, say to terms like h^p. It would then be possible to estimate the asymptotic accuracy of the error as $\mathcal{O}(h^{p+1})$. This would be only for a single step but this theory, if it were feasible, would also give a guide to the global accuracy.

Taylor expansions and trees

Remarkably, $\mathit{flow}_h y_0$ and $\mathcal{RK}_h y_0$ have closely related Taylor expansions, and one of the first aims of this book is to enunciate and analyse these expansions. The first step, in this formulation, is to make use of the graphs known as rooted trees, or arborescences, but referred to throughout this book simply as trees.

The formal introduction of trees will take place in Chapter 2 but, in the meantime, we will introduce these objects by illustrative diagrams:



The set of all trees will be denoted by T.

If t denotes an arbitrary tree, then $|\mathrm{t}|$ is the "order", or number of vertices, and $\sigma(\mathrm{t})$ is the symmetry of t. The symmetry is a positive integer indicating how repetitive a tree diagram is. The formal statement will be given in Definition 2.5A (p. 58).

The common form for $\mathit{flow}_h y_0$ and $\mathcal{RK}_h y_0$ is

$$y_0 + \sum_{\mathrm{t}} \alpha(\mathrm{t}) \tfrac{1}{\sigma(\mathrm{t})} \mathbf{F}(\mathrm{t}) h^{|\mathrm{t}|}, \tag{1.1 c}$$

where, for a given tree, $\mathbf{F}(\mathrm{t})$ depends only on the differential equation being solved and $\alpha(\mathrm{t})$ depends only on the mapping, flow_h or $\mathcal{RK}_h$.

The formulation of various Taylor expansions, given by (1.1 c), is the essential idea behind the theory of B-series, and is the central motivation for this book. We will illustrate the use of this result, using three numerical methods from the present chapter, together with the flow itself. The methods are the Euler method, Euler_h, (1.4 a), the Runge–Kutta method, $\mathit{Runge\text{-}I}_h$:

$$y_1 = y_0 + \tfrac{1}{2}hf(y_0) + \tfrac{1}{2}hf\big(y_0 + hf(y_0)\big)$$

and $\mathit{Runge\text{-}II}_h$, given by (1.1 b). Alternative formulations of these Runge–Kutta methods in Section 1.5 (p. 19) are, for $\mathit{Runge\text{-}I}_h$, (1.5 c) and, for $\mathit{Runge\text{-}II}_h$, (1.5 d).

The coefficients, that is the values of $\Psi(\mathrm{t})$, for $|\mathrm{t}| \leq 3$, are

Mapping	$\Psi(\bullet)$	$\Psi(\mathrm{t}_2)$	$\Psi(\mathrm{t}_3)$	$\Psi(\mathrm{t}_4)$
flow_h	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{6}$
Euler_h	1	0	0	0
$\mathit{Runge\text{-}I}_h$	1	$\frac{1}{2}$	$\frac{1}{2}$	0
$\mathit{Runge\text{-}II}_h$	1	$\frac{1}{2}$	$\frac{1}{4}$	0

Independently of the choice of the differential equation system being solved, we can now state the orders of the three methods under consideration. Because the same entry is given for the single first order tree, each of the three numerical methods is at least first order as an approximation to the exact solution. Furthermore, the two Runge methods are order two but not three, as we see from the agreement with the flow for the order 2 tree, but not for the two order 3 trees. Also, from the table entries, we see that the Euler method does not have an order greater than one.

Fréchet derivatives and gradients

In the formulation and analysis of both initial value problems, and numerical methods for solving them, it will be necessary to introduce various structures involving partial derivatives. In particular, the first Fréchet derivative, also known as the Jacobian matrix, with elements

$$f'(y) = \begin{bmatrix} \frac{\partial f^1}{\partial y^1} & \frac{\partial f^1}{\partial y^2} & \cdots & \frac{\partial f^1}{\partial y^N} \\ \frac{\partial f^2}{\partial y^1} & \frac{\partial f^2}{\partial y^2} & \cdots & \frac{\partial f^2}{\partial y^N} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f^N}{\partial y^1} & \frac{\partial f^N}{\partial y^2} & \cdots & \frac{\partial f^N}{\partial y^N} \end{bmatrix}.$$

Similarly the Fréchet derivative of a scalar-valued function has the form

$$H'(y) = \begin{bmatrix} \frac{\partial H}{\partial y^1} & \frac{\partial H}{\partial y^2} & \cdots & \frac{\partial H}{\partial y^N} \end{bmatrix}.$$

This is closely related to the gradient $\nabla(H) = H'(y)^{\intercal}$ which arises in many specific problems and classes of problems.

Chapter outline

In Section 1.2, a review of differential equations is presented. This is followed in Section 1.3 by examples of differential equations, The Euler and Taylor series methods are introduced in Section 1.4 followed by Runge–Kutta methods (Section 1.5), and multivalue methods (Section 1.6). Finally, a preliminary introduction to B-series is presented in Section 1.7.

1.2 Differential equations

An ordinary differential equation is expressed in the form

$$\frac{\mathrm{d}y}{\mathrm{d}x} = f\big(x, y(x)\big), \qquad f : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N \tag{1.2 a}$$

or, written in terms of individual components,

$$\begin{aligned}
\frac{\mathrm{d}y^1}{\mathrm{d}x} &= f^1\big(x, y^1(x), y^2(x), \ldots, y^N(x)\big),\\
\frac{\mathrm{d}y^2}{\mathrm{d}x} &= f^2\big(x, y^1(x), y^2(x), \ldots, y^N(x)\big),\\
\vdots &\qquad \vdots\\
\frac{\mathrm{d}y^N}{\mathrm{d}x} &= f^N\big(x, y^1(x), y^2(x), \ldots, y^N(x)\big).
\end{aligned} \tag{1.2 b}$$

This can be formulated as an autonomous problem

$$\frac{\mathrm{d}y}{\mathrm{d}x} = f\big(y(x)\big), \qquad f : \mathbb{R}^N \to \mathbb{R}^N, \tag{1.2 c}$$

by increasing N if necessary and introducing a new dependent variable y^0 which is forced to always equal x. This autonomous form of (1.2 b) becomes

$$\begin{aligned}
\frac{\mathrm{d}y^0}{\mathrm{d}x} &= 1,\\
\frac{\mathrm{d}y^1}{\mathrm{d}x} &= f^1\big(y^0(x), y^1(x), y^2(x), \ldots, y^N(x)\big),\\
\frac{\mathrm{d}y^2}{\mathrm{d}x} &= f^2\big(y^0(x), y^1(x), y^2(x), \ldots, y^N(x)\big),\\
\vdots &\qquad \vdots\\
\frac{\mathrm{d}y^N}{\mathrm{d}x} &= f^N\big(y^0(x), y^1(x), y^2(x), \ldots, y^N(x)\big).
\end{aligned}$$

Initial value problems

A subsidiary condition

$$y(x_0) = y_0, \qquad x_0 \in \mathbb{R}, \quad y_0 \in \mathbb{R}^N, \tag{1.2 d}$$

is an initial value and an initial value problem consists of the pair of equations (1.2 a), (1.2 d) or the pair (1.2 c), (1.2 d).

Initial value problems have applications in applied mathematics, engineering, physics and other sciences, and finding reliable and efficient numerical methods for their solution is of vital importance.

Exercise 1 Reformulate the initial value problem

$$\begin{aligned} u''(x) + 3u'(x) &= 2u(x) + v(x) + \cos(x), \quad u(1) = 2, \quad u'(1) = -2, \\ v''(x) + u'(x) - v'(x) &= u(x) + v(x)^2 + \sin(x), \quad v(1) = 1, \quad v'(1) = 4, \end{aligned}$$

in the form $y'(x) = f\big(y(x)\big)$, $y(x_0) = y_0$, where $y^0 = x$, $y^1 = u$, $y^2 = u'$, $y^3 = v$, $y^4 = v'$.

Scalar problems

If $N = 1$, we obtain a scalar initial value problem

$$y'(x) = f\big(x, y(x)\big), \qquad y(x_0) = y_0 \in \mathbb{R}. \tag{1.2 e}$$

Scalar problems are useful models for more general problems, because of their simplicity and ease of analysis. However, this simplicity can lead to spurious conclusions. A specific case is the early analysis of Runge–Kutta order conditions [82] (Runge, 1895), [56] (Heun, 1900), [66] (Kutta, 1901), [77] Nyström, 1925), in which, above order 4, the order conditions derived using (1.2 e) give an incomplete set.

Complex variables

Sometimes it is convenient to write a differential equation using complex variables

$$\frac{\mathrm{d}z}{\mathrm{d}t} = f\big(t, z(t)\big), \qquad f : \mathbb{R} \times \mathbb{C}^N \to \mathbb{C}^N.$$

For example, the system

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= 2x + 3\cos(t), \quad x(0) = 1, \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= 2y + \sin(t), \qquad y(0) = 0, \end{aligned}$$

can be written succinctly as

$$\frac{\mathrm{d}z}{\mathrm{d}t} = 2z + 2\exp(\mathrm{i}t) + \exp(-\mathrm{i}t), \quad z(0) = 1, \tag{1.2 f}$$

with $z(t) = x(t) + \mathrm{i}y(t)$.

Exercise 2 Find the values of A, B, C such that $z = A\exp(2t) + B\exp(\mathrm{i}t) + C\exp(-\mathrm{i}t)$ is the solution to (1.2 f).

Exercise 3 Write the solution to Exercise 2 in terms of the real and imaginary components.

Well-posed problems

An initial value problem is well-posed if it has a solution, this solution is unique and the solution depends continuously on the initial value. In this discussion we will confine ourselves to autonomous problems.

Definition 1.2A A function $f : \mathbb{R}^N \to \mathbb{R}^N$ satisfies a Lipschitz condition if there exists a constant $L > 0$ (the Lipschitz constant) such that

$$\|f(y) - f(z)\| \le L\|y - z\|, \qquad y, z \in \mathbb{R}^N.$$

Given an initial value problem

$$y'(x) = f\big(y(x)\big), \qquad y(x_0) = y_0,$$

where f satisfies a Lipschitz condition with constant L, we find by integration that for $x \ge x_0$,

$$y(x) = y_0 + \int_{x_0}^{x} f\big(y(x)\big)\,\mathrm{d}x. \tag{1.2 g}$$

If $x \in I := [x_0, \bar{x}]$, and $\|y\|$ denotes $\sup_{x\in I} \|y(x)\|$, we can construct a sequence of approximations $y^{[k]}$, $k = 0, 1, \ldots$, to (1.2 g), from

$$\begin{aligned} y^{[0]}(x) &= y_0, \\ y^{[k]}(x) &= y_0 + \int_{x_0}^{x} f\big(y^{[k-1]}(x)\big)\,\mathrm{d}x, \quad k = 1, 2, \ldots. \end{aligned}$$

If $r := |\widehat{x} - x_0|\, L < 1$, we obtain the estimates

$$\|y^{[1]} - y^{[0]}\| \le |\widehat{x} - x_0|\, \|f(y_0)\|, \tag{1.2 h}$$

$$\|y^{[k+1]} - y^{[k]}\| \le r\|y^{[k]} - y^{[k-1]}\| \le r^k |\widehat{x} - x_0|\, \|f(y_0)\|. \tag{1.2 i}$$

This shows that the sequence $y^{[k]}$, $k = 0, 1, \ldots$, is convergent. Denote the limit by y. It can be verified that the conditions for well-posedness are satisfied.

By adding (1.2 h) and (1.2 i), with $k = 1, 2, \ldots$, we see that every member of the sequence satisfies

$$\|y^{[k]} - y^{[0]}\| \le \frac{1}{1 - r} |\widehat{x} - x_0|\, \|f(y_0)\|.$$

To overcome the restriction $|\widehat{x} - x_0|\, L < 1$, a sequence of x values can be inserted between x_0 and $\bar{x}$, sufficiently close together to obtain convergent sequences in each subinterval in turn.

While a Lipschitz condition is very convenient to use in applications, it is not a realistic assumption, because many well-posed problems do not satisfy it. It is perhaps better to use the property given in the following.

Definition 1.2B A function $f : \mathbb{R}^N \to \mathbb{R}^N$ satisfies a local Lipschitz condition if there exists a constant L (the Lipschitz constant) and a positive real R (the influence radius) such that

$$\|f(y) - f(z)\| \leq L\|y - z\|, \qquad y, z \in \mathbb{R}^N, \quad \|y - z\| \leq 2R.$$

If f satisfies the conditions of Definition 1.2B, then for a a given $y_0 \in \mathbb{R}^N$, define a disc D by

$$D = \{y \in \mathbb{R}^N : \|y - y_0\| \leq R\}$$

and a function $\widetilde{f}$ by

$$\widetilde{f}(y) = \begin{cases} f(y), & y \in D, \\ f\big(y_0 + \frac{R}{\|y - y_0\|}(y - y_0)\big), & y \notin D. \end{cases}$$

Exercise 4 Show that $\widetilde{f}$ satisfies a Lipschitz condition with Lipschitz constant L.

The first numerical methods

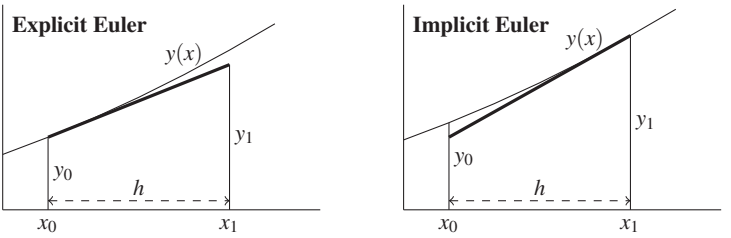
The method of Euler [42] (Euler, Collected works, 1913), proposed in the eighteenth century, is regarded as the foundation of numerical time-stepping methods for the solution of differential equations. We will refer to it here as the "explicit Euler" method to distinguish it from the closely related "implicit Euler" method. Given a problem

$$y'(x) = f\big(x, y(x)\big), \qquad y(x_0) = y_0,$$

we can try to approximate the solution at a nearby point $x_1 = x_0 + h$, by the formula

$$y(x_1) \approx y_1 := y_0 + hf(x_0, y_0).$$

This is illustrated in the one-dimensional case by the diagram on the left (Explicit Euler).



According to this diagram, $y_1 - y_0$ is calculated as the area of the rectangle with width h and height $f(x_0, y_0)$. This is not the correct answer, for which h should be multiplied by the *average* value of $f(y(x))$, but it is often close enough to give useful results for small h. In the diagram on the right (Implicit Euler), the value of $y_1 - y_0$ is h is multiplied by $f(y_1)$, which is not known explicitly but can be evaluated by iteration in the formula

$$y_1 = y_0 + hf(x_1, y_1).$$

We will return to the Euler method in Section 1.4.

1.3 Examples of differential equations

Linear problems

Exponential growth and decay

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \lambda y.$$

If $\lambda > 0$, the solution represents exponential growth and, if $\lambda < 0$, the solution represents exponential decay. Two cases can be combined into a single system

$$\frac{\mathrm{d}}{\mathrm{d}x}\begin{bmatrix} y^1 \\ y^2 \end{bmatrix} = \begin{bmatrix} y^1 \\ -y^2 \end{bmatrix}.$$

This can also be written

$$y' = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \nabla(y^1 y^2)$$

and is an example of a Poisson problem

$$y'(x) = S\nabla(H(y)), \tag{1.3 a}$$

where S is a skew-symmetric matrix. For such problems $H(y(x))$ has a constant value, because

$$\frac{\mathrm{d}H\big(y(x)\big)}{\mathrm{d}x} = \Big(\frac{\partial H}{\partial y}\Big) S \Big(\frac{\partial H}{\partial y}\Big)^{\intercal} = 0.$$

It is an important aim in numerical analysis to preserve this invariance property, in computational results.

A four-dimensional linear problem

The problem

$$y' = My, \qquad M = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -2 \end{bmatrix},$$

is a trivial special case of the discretized diffusion equation on an interval domain. A transformation $M \to \widehat{M} = T^{-1}MT$, where

$$T = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}, \qquad \widehat{M} = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -3 & 1 \\ 0 & 0 & 1 & -2 \end{bmatrix},$$

partitions the problem into symmetric and anti-symmetric components. Also write $\widehat{y} = T^{-1}y$, $\widehat{y}_0 = T^{-1}y_0$ so that the partitioned initial value problem becomes

$$\widehat{y}' = \widehat{M}\widehat{y}, \qquad \widehat{y}(x_0) = \widehat{y}_0.$$

Making this transformation converts the problem into two separate two dimensional problems which can be solved independently and the results recombined.

Harmonic oscillator and simple pendulun

The harmonic oscillator:

$$\frac{\mathrm{d}}{\mathrm{d}x}\begin{bmatrix} y^1 \\ y^2 \end{bmatrix} = \begin{bmatrix} y^2 \\ -y^1 \end{bmatrix}.$$

This equation can be recast in scalar complex form by introducing a new variable $z = y^1 + \mathrm{i}y^2$. It then becomes

$$\frac{\mathrm{d}z}{\mathrm{d}x} = -\mathrm{i}z.$$

The harmonic oscillator can also be written in the form (1.3 a), with

$$H(y) = \tfrac{1}{2}\left((y^1)^2 + (y^2)^2\right).$$

The simple pendulum:

$$\frac{\mathrm{d}}{\mathrm{d}x}\begin{bmatrix} y^1 \\ y^2 \end{bmatrix} = \begin{bmatrix} y^2 \\ -\sin(y^1) \end{bmatrix}.$$

This problem is not linear but, if $\|y(0)\|$ is sufficiently small, the simple pendulum is a reasonable approximation to a linear problem, because $\sin(y^1) \approx y^1$. It also has the form of (1.3 a) with $H(y) = \frac{1}{2}(y^2)^2 - \cos(y^1)$.

Stiff problems

Many problems arising in scientific modelling have a special property known as "stiffness", which makes numerical solution by classical methods very difficult. An early reference is [35] (Curtiss, Hirschfelder,1952). For a contemporary study of stiff problems, and numerical methods for their solution, see [53] (Hairer, Nørsett, Wanner,1993) and [86] (Söderlind, Jay, Calvo, 2015).

When attempting to determine the most appropriate stepsize to use with a particular method, and a particular problem, many considerations come into play. The first is the requirement that the truncation error is sufficiently small to match the requirements of the physical application, and the second is that the numerical results are not corrupted unduly by unstable behaviour.

To illustrate this idea, consider the use of the Euler method (see Section 1.4 (p. 14)), applied to the three-dimensional problem

$$\frac{\mathrm{d}}{\mathrm{d}x}\begin{bmatrix} y^1 \\ y^2 \\ y^3 \end{bmatrix} = \begin{bmatrix} -y^2 + 0.40001(y^3)^2 \\ y^1 \\ -100y^3 \end{bmatrix}, \quad \begin{bmatrix} y^1(0) \\ y^2(0) \\ y^3(0) \end{bmatrix} = \begin{bmatrix} 0.998 \\ 0.00001 \\ 1 \end{bmatrix}, \qquad (1.3\,\mathrm{b})$$

with exact solution

$$\begin{bmatrix} y^1(x) \\ y^2(x) \\ y^3(x) \end{bmatrix} = \begin{bmatrix} \cos(x) - 0.002\exp(-200x) \\ \sin(x) + 0.00001\exp(-200x) \\ \exp(-100x) \end{bmatrix}.$$

A solution by the Euler method consists of computing approximations

$$y_1 \approx y(x_0 + h), \quad y_2 \approx y(x_0 + 2h), \quad y_3 \approx y(x_0 + 3h), \quad \ldots,$$

using $y_n = F(y_{n-1})$, $n = 1, 2, \ldots$, where

$$F(u) = \begin{bmatrix} u^1 + h\big(-u^2 + 0.40001(u^3)^2\big) \\ u^2 + hu^1 \\ (1 - 100h)u^3 \end{bmatrix}.$$

For sequences like this, stability, for the third component, depends on the condition $1 - 100h \geq -1$ being satisfied, so that $h \leq 0.02$. If this condition is not satisfied, unstable behaviour of y^3 will feed into the first two components and the computed

results cannot be relied on. However, if the initial value for y^3 were zero, and this component never drifted from this value, there would be no such restriction on obtaining reliable answers.

Exercise 5 If problem (1.3 b) is solved using the implicit Euler method (1.4 d), find the function $\widehat{F}$ such that $y_n = \widehat{F}(y_{n-1})$, and show that there is no restriction on positive h to yield stable results.

Test problems

A historical problem

The following one-dimensional non-autonomous problem was used by Runge and others to verify the behaviour of some early Runge–Kutta methods:

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{y-x}{y+x}, \qquad y(0) = 1. \tag{1.3 c}$$

A parametric solution $t \mapsto \big(y(t), x(t)\big) := \big(y^1(t), y^2(t)\big)$ can be found from the system

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y^1 \\ y^2 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} y^1 \\ y^2 \end{bmatrix}, \qquad \begin{bmatrix} y^1(0) \\ y^2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

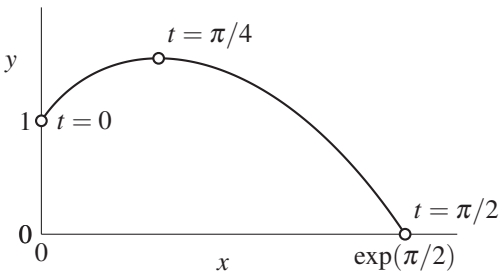
and, by writing $z = y^1 + \mathrm{i}y^2$, we obtain

$$\frac{\mathrm{d}z}{\mathrm{d}t} = (1+\mathrm{i})z, \qquad z(0) = 1,$$

with solution $z = \exp\big((1+\mathrm{i})t\big)$, so that, reverting to the original notation,

$$y(t) = \exp(t)\cos(t),$$
$$x(t) = \exp(t)\sin(t).$$

The solution on $\big[0, \exp(\frac{1}{2}\pi)\big]$ corresponds to $t \in \big[0, \frac{1}{2}\pi\big]$ and is shown in the diagram

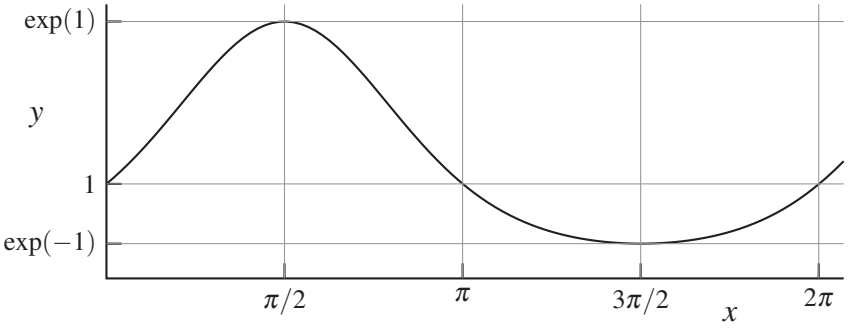


A problem from DETEST

One of the pioneering developments in the history of numerical methods for differential equations is the use of standardized test problems. These have been useful in identifying reliable and accurate software. This problem from the DETEST set [57] (Hull, Enright, Fellen, Sedgwick, 1972) is an interesting example.

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \cos(x)y, \quad y(0) = 1.$$

The exact solution, given by $y = \exp\big(\sin(x)\big)$, is shown in the diagram



The Prothero–Robinson problem

The problem of Prothero and Robinson [79] (1974),

$$y'(x) = g'(x) + L\big(y - g(x)\big),$$

where $g(x)$ is a known function, was introduced as a model for studying the behaviour of numerical methods applied to stiff problems. A special case is

$$y' = \cos(x) - 10\big(y - \sin(x)\big), \qquad y(0) = 0,$$

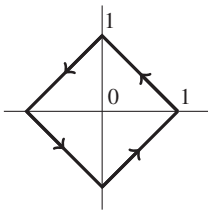
with general solution $y(x) = C\exp(-10x) + \sin(x)$, where $C = 0$ when $y(0) = 0$.

A problem with discontinuous derivatives

The two-dimensional "diamond problem", as we will name it, is defined to have piecewise constant derivative values which change from quadrant to quadrant as follows

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \begin{cases} \begin{bmatrix} -1 \\ 1 \end{bmatrix}, & y^1 > 0, \quad y^2 \geq 0, \\ \begin{bmatrix} -1 \\ -1 \end{bmatrix}, & y^1 \leq 0, \quad y^2 > 0, \\ \begin{bmatrix} 1 \\ -1 \end{bmatrix}, & y^1 < 0, \quad y^2 \leq 0, \\ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, & y^1 \geq 0, \quad y^2 < 0. \end{cases}$$

Using the initial value $y = \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathsf{T}}$, the orbit, with period 4, is as in the diagram:



This problem is interesting as a numerical test because of the non-smoothness of the orbit as it moves from one quadrant to the next.

The Kepler problem

$$\frac{\mathrm{d}}{\mathrm{d}x} \begin{bmatrix} y^1 \\ y^2 \\ y^3 \\ y^4 \end{bmatrix} = \begin{bmatrix} y^3 \\ y^4 \\ -\dfrac{y^1}{r^3} \\ -\dfrac{y^2}{r^3} \end{bmatrix}, \tag{1.3 d}$$

where $r = \left((y^1)^2 + (y^2)^2\right)^{1/2}$. The Kepler problem satisfies conservation of energy $H' = 0$, where

$$H(x) = \tfrac{1}{2}\left((y^3)^2 + (y^4)^2\right) - r^{-1}$$

and also conservation of angular momentum $A' = 0$, where

$$A(x) = y^1 y^4 - y^2 y^3.$$

Exercise 6 Show that $H(x)$ is invariant.

Exercise 7 Show that $A(x)$ is invariant.

1.4 The Euler method

The explicit Euler method as a Taylor series method

Given a differential equation and an initial value,

$$y'(x) = f(x,y), \qquad y(x_0) = y_0,$$

the Taylor series formula is a possible approach to finding an approximation to $y(x_0+h)$:

$$y(x_0+h) \approx y(x_0) + hy'(x_0) + \tfrac{1}{2!}h^2y''(x_0) + \cdots + \tfrac{1}{p!}h^p y^{(p)}(x_0).$$

If y is a sufficiently smooth function, then we would expect the error in this approximation to be $O(h^{p+1})$. When $p = 1$, this reduces to the Euler method. This is very convenient to use, because both $y(x_0) = y_0$ and $y'(x_0) = f(x_0, y_0)$ are known in advance. However, for $p = 2$, we would need the value of $y''(x_0)$, which can be found from the chain rule:

$$y''(x) = \frac{d}{dx} f\big(x, y(x)\big) = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{dy}{dx} = f_x + f_y f,$$

where the subscripts in f_x and f_y denote partial derivatives, and, for brevity, the arguments have been suppressed. Restoring the arguments, we can write

$$y''(x_0) = f_x(x_0,y_0) + f_y(x_0,y_0)f(x_0,y_0).$$

The increasingly more complicated expressions for $y^{(3)}$, $y^{(4)}$, …, have been worked out at least to order 6 [59] (Hut'a, 1956), and they are summarized here to order 4.

$$\begin{aligned}
y' &= f,\\
y'' &= f_x + f_y f,\\
y^{(3)} &= f_{xx} + 2f_{xy}f + f_{yy}f^2 + f_x f_y + f_y^2 f,\\
y^{(4)} &= f_{xxx} + 3f_{xxy}f + 3f_{xy}f_x + 5f_{xy}f_y f + 3f_{xyy}f^2 + f_y f_{xx}\\
&\qquad + 3f_x f_{yy} f + f_y^2 f_x + f_y^3 f + 4f_y f^2 f_{yy} + f^3 f_{yyy}.
\end{aligned}$$

We will return to the evaluation of higher derivatives, in the case of an autonomous system, in Section 1.7 (p. 33).

Exercise 8 Given the differential equation $y' = y + \sin(x)$, find $y^{(n)}$ for $n \le 7$.

The explicit Euler method

The Euler method produces the result

$$y_k = y_{k-1} + hf(x_{k-1}, y_{k-1}), \qquad k = 1,2,\ldots. \tag{1.4 a}$$

In this introduction, it will be assumed that h is constant. Now consider a numerical method of the form

$$y_k = y_{k-1} + h\Psi(x_{k-1}, y_{k-1}), \tag{1.4 b}$$

used in the same way as the Euler method.

Definition 1.4A The method defined by (1.4 b) is convergent if, for a problem defined by $f(x,y)$, $y(x_0) = y_0$, with the solution Y_n, at $\bar{x}$, approximated using n steps with $h = (\bar{x} - x_0)/n$, then

$$\lim_{n\to\infty} Y_n = y(\bar{x}).$$

Theorem 1.4B The Euler method is convergent.

This result from [36] (Dahlquist, 1956), with an exposition in the classic textbook [55] (Henrici, 1962), is also presented in the more recent books [50] (Hairer, Nørsett, Wanner, 1993) and [20] (Butcher, 2016).

Variable stepsize

The standard formulation of a one-step method is based on a single input y_0, and its purpose is to calculate a single output y_1. However, it is also possible to consider the input as being the pair $[y_0, hf_0]$, with $f_0 = f(y_0)$. In this case the output would be a pair $[y_1, hf_1]$. Apart from the inconvenience of passing additional data between steps, the two formulations are identical.

However, the two input approach has an advantage if the Euler method is required to be executed as a variable stepsize method, as in the Octave function (1.4 c).As we will see in Section 1.5, the Runge–Kutta method (1.5 c) has order 2. This would mean that half the difference between the result computed by Euler, and the result computed by this particular Runge–Kutta method, could be used as an error estimator for the Euler result because $y_0 + \frac{1}{2}hf_0 + \frac{1}{2}hf_1$ is identical to the result computed by (1.5 c). This is the basis for the function represented in (1.4 c). Note that this estimation does not require additional f calculations.

```
function [yout,hfyout,hout] = Euler(y,hfy,tolerance)
   yout = y + hfy;
   hfyout = h * f(yout);
   error = 0.5 * norm(hfy - hfyout);
   r = sqrt(tolerance / error);
   hout = r * h;
   hfyout = r * hfyout;
endfunction
```
(1.4 c)

Exercise 9 Discuss the imperfections in (1.4 c).

The implicit Euler method

As we saw in Section 1.3, through the problem (1.3 b), there are sometimes advantages in using the implicit version of (1.4 a), in the form

$$y_k = y_{k-1} + hf(x_k, y_k), \qquad k = 1, 2, \ldots \tag{1.4 d}$$

This method also reappears as an example of the implicit theta Runge–Kutta method (1.5 g) with $\theta = 1$.

In the calculation of y_k in (1.4 d), we need to solve an algebraic equation

$$Y - hf(Y) = C, \text{ where } C = y_{k-1}.$$

If f satisfies a Lipschitz condition with $|h|L < 1$, then it is possible to use functional iteration. That is, Y can be found numerically from the sequence $Y^{[0]}, Y^{[1]}, Y^{[2]}, \ldots$, where

$$\begin{aligned} Y^{[0]} &= C, \\ Y^{[n]} &= C + hf(Y^{[n-1]}), \qquad n = 1, 2, \ldots \end{aligned}$$

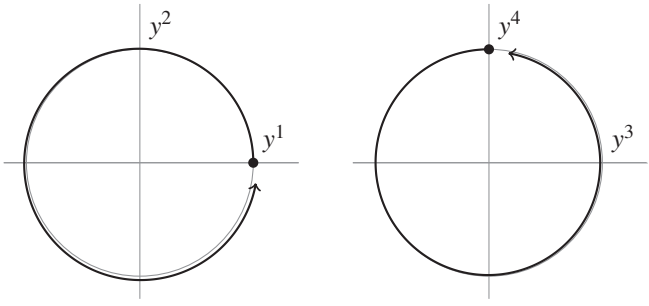
To obtain rapid convergence, this simple iterative system can be replaced by the quadratically-convergent Newton scheme:

$$\begin{aligned} Y^{[0]} &= C, \\ Y^{[n]} &= Y^{[n-1]} - \left(I - hf'(Y^{[n-1]})\right)^{-1} \left(Y^{[n-1]} - C - hf(Y^{[n-1]})\right), \quad n = 1, 2, \ldots \end{aligned}$$

Experiments with the explicit Euler method

The Kepler problem

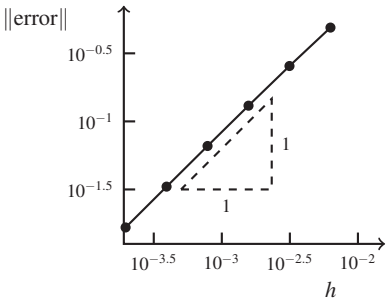
The Kepler problem (1.3 d), with initial value $y_0 = [1, 0, 0, 1]^{\mathsf{T}}$, has a circular orbit solution with period 2π. To see how well the Euler method is able to solve this problem over a single orbit, a constant stepsize $h = 2\pi/n$ is used over n steps in each of the cases $n = 1000 \times 2^k$, $k = 0, 1, \ldots, 5$. As a typical case, $n = 2000$ is shown in the following diagrams, where the first and second components are shown in the left-hand diagram, and the third and fourth components on the right:



To assess the accuracy, in each of the six cases, it is convenient to calculate $\|y_n - y_0\|_2$. For example, if $n = 1000$, then

$$\begin{aligned} y_n &= [1.015572, -0.358194, 0.319112, 0.907596]^\intercal, \\ y_n - y_0 &= [-0.015572, 0.358194, -0.319112, 0.092404]^\intercal, \\ \|y_n - y_0\|_2 &= 0.488791. \end{aligned}$$

This single result gives only limited information on the accuracy to be expected from the Euler method when carrying out this type of calculation. It will be more interesting to use the sequence of six n values, $n = 1000, 2000, \ldots .32000$, with corresponding stepsizes $h = 2\pi/1000, 2\pi/2000, \ldots, 2\pi/32000$, displayed in a single diagram. As we might expect, the additional work as n doubles repeatedly gives systematic improvements. To illustrate the behaviour of this calculation for increasingly high values of n, and increasingly low values of h, the following diagram is presented



The triangle shown beside the main line suggests that the slope is close to 1.

The slope of lines relating error to stepsize is of great importance since it predicts the behaviour that could be expected for extremely small h. For example, if we needed 10^{-6} accuracy this figure suggests that we would need a stepsize of about 10^{-8} and this would require a very large number of steps and therefore an unreasonable amount of computer time. If, on the other hand, the slope were 2 or greater, we would obtain much better performance for small h.

Experiments with diamond

In the case of the diamond problem, it is possible to evaluate the accumulated error in a single orbit, evaluated using the Euler method. If n, the number of steps to be evaluated, is a multiple of 4, there is zero error. We will consider the case $n = 4m + k$, with $m + k \geq 4$, where $1 \leq k \leq 3$. Because the period is 4, the stepsize is $h = 4/n$. In the first quadrant, $m + 1$ steps moves the solution to the second quadrant and a further $m + 1$ advances the solution to the interface with the third quadrant. It then takes $m + 2$ steps to move to the fourth quadrant. This leaves $4m + k - 2(m + 1) - (m + 2) = m - (4 - k)$ steps to move within the fourth quadrant. The final position, relative to the initial point, is then

$$\frac{m+1}{n/4}\begin{bmatrix}-1\\1\end{bmatrix}+\frac{m+1}{n/4}\begin{bmatrix}-1\\-1\end{bmatrix}+\frac{m+2}{n/4}\begin{bmatrix}1\\-1\end{bmatrix}+\frac{m+k-4}{n/4}\begin{bmatrix}1\\1\end{bmatrix}=\frac{4}{n}\begin{bmatrix}k-4\\k-6\end{bmatrix}.$$

Computer simulations for this calculation can be misleading because of round-off error.

Exercise 10 Find the error in calculating two orbits of diamond using $n = 8m+k$ steps with $1 \le k \le 7$, with m sufficiently large.

An example of Taylor series

From the many choices available to test the Taylor series method, we will look at the initial value problem

$$y' = x^2 + y^2, \qquad y(0) = 1. \tag{1.4e}$$

In [55] (Henrici, 1962), this problem was used to illustrate the disadvantages of Taylor series methods, because of rapid growth of the complexity of the formulae for $y'', y^{(3)}, \ldots$. This was in the relatively early days of digital computing, and the situation has now changed because of the feasibility of evaluating Taylor terms automatically.

But going back to hand calculations, the higher-derivatives do indeed blossom in complexity, as we see from the first few members of the sequence

$$\begin{aligned}
y' &= x^2 + y^2,\\
y'' &= 2x + 2x^2y + 2y^3,\\
y^{(3)} &= 2 + 4xy + 2x^4 + 8x^2y^2 + 6y^4,\\
y^{(4)} &= 4y + 12x^3 + 20xy^2 + 20x^4y + 40x^2y^3 + 24y^5.
\end{aligned}$$

Recursive computation of derivatives

Although we will not discuss the systematic evaluation of higher derivatives for a general problem, we can at least find a simple recursion for the example problem (1.4e), based on the formula

$$y^{(n)} = \frac{\partial}{\partial x}y^{(n-1)} + \frac{\partial}{\partial y}y^{(n-1)}f.$$

This gives the sequence of formulae

$$\begin{aligned}
y^{(1)} &= x^2 + (y^{(0)})^2,\\
y^{(2)} &= 2x + 2y^{(0)}y^{(1)},\\
y^{(3)} &= 2 + 2y^{(0)}y^{(2)} + 2(y^{(1)})^2,\\
y^{(4)} &= 2y^{(0)}y^{(3)} + 6y^{(1)}y^{(2)},\\
y^{(5)} &= 2y^{(0)}y^{(4)} + 8y^{(1)}y^{(3)} + 6(y^{(2)})^2,\\
y^{(6)} &= 2y^{(0)}y^{(5)} + 10y^{(1)}y^{(4)} + 20y^{(2)}y^{(3)},\\
y^{(7)} &= 2y^{(0)}y^{(6)} + 12y^{(1)}y^{(5)} + 30y^{(2)}y^{(4)} + 20(y^{(3)})^2,
\end{aligned}$$

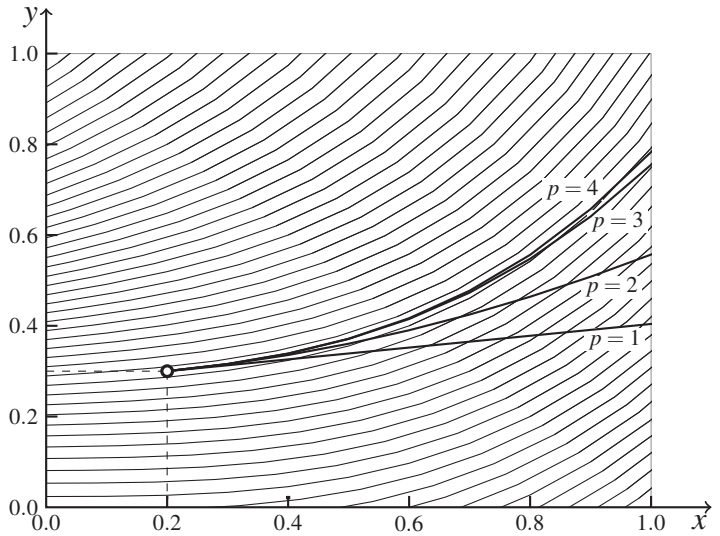


Figure 1 Taylor series approximations of orders $p = 1,2,3,4$ for $y' = x^2 + y^2$, $y(0.2) = 0.3$

and the general result

$$y^{(n)} = \sum_{i=0}^{n-1} \binom{n-1}{i} y^{(i)} y^{(n-1-i)}, \qquad n \geq 4.$$

To demonstrate how well the Taylor series works for this example problem, Figure 1 is presented.

1.5 Runge–Kutta methods

One of the most widely used families of methods for approximating the solutions of differential equations is the Runge–Kutta family. In one of these methods, a sequence of n steps is taken from an initial point, x_0, to obtain an approximation to the solution at $x_0 + nh$, where h is the "stepsize".

Each step has the same form and we will consider only the first. Write the input approximation as $y_0 \approx y(x_0)$. The method involves first obtaining approximations $Y_i \approx y(x_0 + hc_i)$, $i = 1,2,\ldots,s$, where $c_1, c_2, \ldots, c_s$ are the stage abscissae. Write $F_i = f(Y_i)$ for each stage so that $F_i \approx y'(x_0 + hc_i)$. The actual approximations used for the stage values take the form

$$Y_i = y_0 + h \sum_{j<i} a_{ij} F_j, \qquad i = 1,2,\ldots,s. \tag{1.5 a}$$

After the stage values, $Y_1, Y_2, \ldots$, and the stage derivatives, $F_1, F_2, \ldots$, have been evaluated, the output to the step is found from

$$y_1 = y_0 + h\sum_{i=1}^{s} b_i F_i. \tag{1.5 b}$$

Examples of explicit Runge–Kutta methods

The Runge second order methods

The method *Runge-I* is defined by the equations

$$\begin{aligned} Y_1 &= y_0, & F_1 &= f(x_0, Y_1),\\ Y_2 &= y_0 + hF_1, & F_2 &= f(x_0 + h, Y_2),\\ y_1 &= y_0 + \tfrac{1}{2}(hF_1 + hF_2). \end{aligned} \tag{1.5 c}$$

Because $F_1 \approx y'(x_0)$ and $F_2 \approx y'(x_1)$, (1.5 c) can be seen as a generalization of the trapezoidal rule:

$$y(x_0 + h) - y(x_0) \approx \tfrac{1}{2}\big(hy'(x_0) + hy'(x_0 + h)\big).$$

The method *Runge-II* is defined by the equations

$$\begin{aligned} Y_1 &= y_0, & F_1 &= f(x_0, Y_1),\\ Y_2 &= y_0 + \tfrac{1}{2}hF_1, & F_2 &= f(x_0 + \tfrac{1}{2}h, Y_2),\\ y_1 &= y_0 + hF_2. \end{aligned} \tag{1.5 d}$$

Because $F_1 \approx y'(x_0)$ and $F_2 \approx y'(x_0 + \frac{1}{2}h)$, (1.5 c) can be seen as a generalization of the midpoint rule:

$$y(x_0 + h) - y(x_0) \approx hy'(x_0 + \tfrac{1}{2}h).$$

Third and fourth order methods

There are many possible methods with three stages and order three, and the following is an example:

$$\begin{aligned} Y_1 &= y_0, & F_1 &= f(x_0, Y_1),\\ Y_2 &= y_0 + \tfrac{1}{3}hF_1, & F_2 &= f(x_0 + \tfrac{1}{3}h, Y_2),\\ Y_3 &= y_0 + \tfrac{2}{3}hF_2, & F_3 &= f(x_0 + \tfrac{2}{3}h, Y_3),\\ y_1 &= y_0 + \tfrac{1}{4}hF_1 + \tfrac{3}{4}hF_3. \end{aligned} \tag{1.5 e}$$

Similarly, the following four stage fourth order method is one of a large family:

$$\begin{aligned} Y_1 &= y_0, & F_1 &= f(x_0, Y_1),\\ Y_2 &= y_0 + \tfrac{1}{4}hF_1, & F_2 &= f(x_0 + \tfrac{1}{4}h, Y_2),\\ Y_3 &= y_0 + \tfrac{1}{2}hF_2, & F_3 &= f(x_0 + \tfrac{1}{2}h, Y_3),\\ Y_4 &= y_0 + hF_1 - 2hF_2 + 2hF_3, & F_4 &= f(x_0 + h, Y_4),\\ y_1 &= y_0 + \tfrac{1}{6}hF_1 + \tfrac{2}{3}hF_3 + \tfrac{1}{6}hF_4. \end{aligned} \tag{1.5 f}$$

Naive verification of order

Although the criteria for order of a Runge–Kutta method are quite sophisticated, it is possible to demonstrate why (1.5 c) and (1.5 d) each has order 2, using very simple arguments. We will assume that f is a sufficiently smooth function so that we can always use Taylor series in the form

$$y(x_0+h) = y(x_0) + \sum_{i=1}^{n} \frac{h^i}{i!} y^{(i)}(x_0) + \mathcal{O}(h^{n+1}).$$

Thus for the method (1.5 c), we can write for the truncation error of Y_2, as an approximation to $y(x_0+h)$,

$$y(x_0+h) - Y_2 = y(x_0+h) - y(x_0) - hy'(x_0) = \tfrac{1}{2}h^2 y''(x_0) + \mathcal{O}(h^3).$$

Assuming the existence and smoothness of f_y, we can also write

$$\begin{aligned} y'(x_0+h) - F_2 &= f(x_0+h, y(x_0+h)) - f(x_0+h, Y_2) \\ &= \tfrac{1}{2}h^2 f_y(x_0,y_0)y''(x_0) + \mathcal{O}(h^3) \end{aligned}$$

For the truncation error in y_1, as an approximation to $y(x_0+h)$, we have

$$\begin{aligned} & y(x_0+h) - y(x_0) - \tfrac{1}{2}hF_1 - \tfrac{1}{2}hF_2 \\ &\quad = \big(y(x_0+h) - y(x_0) - \tfrac{1}{2}hy'(x_0) - \tfrac{1}{2}hy'(x_0+h)\big) \\ &\qquad\qquad + \tfrac{1}{2}h\big(y'(x_0+h) - F_2\big) \\ &\quad = \big(hy'(x_0) + \tfrac{1}{2}h^2y''(x_0) + \tfrac{1}{6}h^3y^{(3)}(x_0) \\ &\qquad - \tfrac{1}{2}hy'(x_0) - \tfrac{1}{2}hy'(x_0) - \tfrac{1}{2}h^2y''(x_0) - \tfrac{1}{4}h^3y^{(3)}(x_0)\big) \\ &\qquad\qquad + \tfrac{1}{2}h\big(\tfrac{1}{2}h^2 f_y(x_0,y_0)y''(x_0)\big) + \mathcal{O}(h^4) \\ &\quad = -\tfrac{1}{12}h^3y^{(3)}(x_0) + \tfrac{1}{4}h^3 f_y(x_0,y_0)y''(x_0) + \mathcal{O}(h^4). \end{aligned}$$

Exercise 11 Find a similar error formula for (1.5 d).

Exercise 12 Show that the method (1.5 e) has order 3.

Exercise 13 Show that the method (1.5 f) has order 4.

Representing methods with tableaux

It is customary to represent a particular Runge–Kutta method using only the coefficients a_{ij}, b_i, c_i appearing in (1.5 a, 1.5 b). These are, for the classical explicit case, arranged in a tableau as follows

$$\begin{array}{c|cccccc} 0 & & & & & & \\ c_2 & a_{21} & & & & & \\ c_3 & a_{31} & a_{32} & & & & \\ \vdots & \vdots & \vdots & \ddots & & & \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & & \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array} .$$

For the methods we have already introduced, the corresponding tableaux are

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \qquad \text{method (1.5 c)}$$

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array} \qquad \text{method (1.5 d)}$$

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array} \qquad \text{method (1.5 e)}$$

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{4} & \frac{1}{4} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 1 & -2 & 2 & \\ \hline & \frac{1}{6} & 0 & \frac{2}{3} & \frac{1}{6} \end{array} \qquad \text{method (1.5 f)}$$

Implicit Runge–Kutta methods

If the coefficient matrix is full — that is, it contains non-zero elements on and above the diagonal — the stages cannot be computed sequentially, and in order, using explicit computations. Hence, an iteration scheme is normally required for their evaluation. For example, the "theta methods" with tableaux of the form

$$\begin{array}{c|cc} \theta & \theta & \\ \hline & 1 & \end{array} \qquad (1.5\,\text{g})$$

are explicit only if $\theta = 0$. Two important special cases are $\theta = \frac{1}{2}$ (the implicit mid-point rule method) and $\theta = 1$ (the implicit Euler method). If f is sufficiently smooth and h is sufficiently small, then the single stage Y is to be a solution of $Y = y_0 + h\theta f(Y)$ and can be evaluated by functional iteration:

$$\begin{aligned} Y^{[0]} &= y_0, \\ Y^{[k]} &= y_0 + h\theta f\big(Y^{[k-1]}\big), \qquad k = 1, 2, \ldots \end{aligned}$$

For many problems, this iteration scheme is not efficient, because of the severe limitation that might need to be imposed on $|h|$, and some variant of Newton iteration must be used.

For $s = 2$, a well-known example of an implicit method is the so-called Radau IIA method, with order 3, given by the tableau

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array} \tag{1.5 h}$$

A second fully-implicit method with $s = 2$ is known as a Gauss method and has order $p = 4$ [54] (Hammer, Hollingsworth,1955). The tableau is

$$\begin{array}{c|cc} \frac{1}{2} - \frac{1}{6}\sqrt{3} & \frac{1}{4} & \frac{1}{4} - \frac{1}{6}\sqrt{3} \\ \frac{1}{2} + \frac{1}{6}\sqrt{3} & \frac{1}{4} + \frac{1}{6}\sqrt{3} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} . \tag{1.5 i}$$

This is one of a family of methods based on Gaussian quadrature and with order $p = 2s$ [65] (Kuntzmann, 1961), [9] (Butcher, 1964).

Inverse and adjoint methods

The stages and final output for a generic Runge–Kutta method, assuming input value y_0, are given by

$$Y_i = y_0 + h \sum_{j=1}^{s} a_{ij} f(Y_j), \quad i = 1, 2, \ldots, s, \tag{1.5 j}$$

$$y_1 = y_0 + h \sum_{j=1}^{s} b_j f(Y_j). \tag{1.5 k}$$

If y_1 is already known, y_0 can be found by solving from (1.5 k), and the Y_i can be found by subtracting (1.5 k) from (1.5 j). This gives the method

$$Y_i = y_1 + h \sum_{j=1}^{s} (a_{ij} - b_j) f(Y_j), \quad i = 1, 2, \ldots, s,$$

$$y_0 = y_1 + h \sum_{j=1}^{s} (-b_j) f(Y_j),$$

which exactly undoes the work of the original method. This leads to the definition

Definition 1.5A Given a tableau

$$\begin{array}{c|c} c & A \\ \hline & b^{\intercal} \end{array}$$

the inverse method (inverse tableau) is

$$\begin{array}{c|c} c-(b^{\intercal}\mathbf{1})\mathbf{1} & A-\mathbf{1}b^{\intercal} \\ \hline & -b^{\intercal} \end{array}$$

Closely related are "adjoint methods" in which the sign of h is changed in an inverse method. For example, the adjoint method of (1.5 i) is

$$\begin{array}{c|cc} \frac{1}{2}+\frac{1}{6}\sqrt{3} & \frac{1}{4} & \frac{1}{4}+\frac{1}{6}\sqrt{3} \\ \frac{1}{2}-\frac{1}{6}\sqrt{3} & \frac{1}{4}-\frac{1}{6}\sqrt{3} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

which becomes identical with (1.5 i) if the stages are numbered in reverse order. Methods with this property are "self adjoint" and have important properties computationally.

Methods with general index sets

The flow of an autonomous initial value problem on the interval $[0,1]$ can be written as the solution to the integral equation

$$\begin{aligned} y(x_0+\xi h) &= y_0 + h\int_0^{\xi} f\big(y(x_0+\eta h)\big)\,\mathrm{d}\eta \\ &= y_0 + h\int_0^{1} H(\xi-\eta) f\big(y(x_0+\eta h)\big)\,\mathrm{d}\eta, \end{aligned}$$

where

$$H(x) = \begin{cases} 0, & x<0, \\ \frac{1}{2}, & x=0, \\ 1, & x>0, \end{cases}$$

denotes the Heavyside function.

This can be regarded as the continuous analogue of the s-stage Runge–Kutta method with the coefficient matrix given by

$$a_{ij} = \begin{cases} 0, & i < j, \\ \frac{1}{2s}, & i = j, \\ \frac{1}{s}, & i > j. \end{cases}$$

It is possible to place these two methods on a common basis by introducing an "index set" I [14] (Butcher, 1972) which, in these examples, could be $[0,1]$ or $\{1,2,3,\ldots,s\}$. Adapting Runge–Kutta terminology slightly, the stage value function becomes a bounded mapping $I \to \mathbb{R}^N$ and the coefficient matrix A becomes a bounded linear operator on the space of bounded mappings $I \to \mathbb{R}$ to this same space. The final component of a Runge–Kutta method specification, that is the row vector $b^\intercal$, becomes a linear functional on the bounded mappings $I \to \mathbb{R}$. More details will be presented in Chapter 4.

Even though energy-preserving Runge–Kutta methods, with finite I, do not exist, the following method, the "Average Vector Field" method [80] (Quispel, McLaren, 2008)) does satisfy this requirement [29] (Celledoni et al, 2009).

$$y_1 = y_0 + h\int_0^1 f\big((1-\eta)y_0 + \eta y_1\big)\,\mathrm{d}\,\eta.$$

For this method we have

$$\begin{aligned} I &= [0,1], \\ A(\xi)\phi &= \xi\int_0^1 \phi(\eta)\,\mathrm{d}\,\eta, \\ b^\intercal\phi &= \int_0^1 \phi(\eta)\,\mathrm{d}\,\eta. \end{aligned}$$

Methods based on the index set $[0,1]$ are referred to as "Continuous stage Runge–Kutta methods".

Equivalence classes of Runge–Kutta methods

The two Runge–Kutta methods

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & 0 & 1 & \\ \hline & 0 & 1 & 0 \end{array}, \qquad \begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \hline & 0 & 1 \end{array}$$

are equivalent in the sense that they give identical results because the third stage of the method on the left is evaluated and never used. This is an example of Dahlquist–Jeltsch equivalence [39] (Dahlquist, Jeltsch, 2006). Similarly the two implicit methods

$$\begin{array}{c|cc} \frac{1}{2}-\frac{1}{6}\sqrt{3} & \frac{1}{4} & \frac{1}{4}-\frac{1}{6}\sqrt{3} \\ \frac{1}{2}+\frac{1}{6}\sqrt{3} & \frac{1}{4}+\frac{1}{6}\sqrt{3} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}, \qquad \begin{array}{c|cc} \frac{1}{2}+\frac{1}{6}\sqrt{3} & \frac{1}{4} & \frac{1}{4}+\frac{1}{6}\sqrt{3} \\ \frac{1}{2}-\frac{1}{6}\sqrt{3} & \frac{1}{4}-\frac{1}{6}\sqrt{3} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

are equivalent because they are the same method with their stages numbered in a different order.

Another example of an equivalent pair of methods, is

$$\begin{array}{c|cccc} \frac{1}{3} & \frac{1}{3} & \frac{1}{12} & -\frac{1}{12} & 0 \\ \frac{1}{3} & -\frac{1}{12} & \frac{1}{2} & \frac{1}{12} & -\frac{1}{6} \\ 1 & \frac{1}{2} & \frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{8} & \frac{1}{8} \\ \hline & \frac{3}{8} & \frac{3}{8} & \frac{1}{3} & -\frac{1}{12} \end{array}, \qquad \begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}.$$

Suppose Y_1^*, Y_2^* are the solutions computed using the method on the right. Then $Y_1 = Y_2 = Y_1^*$ and $Y_3 = Y_4 = Y_2^*$ satisfy the stage conditions for the method on the left. Hence, the outputs for each of the methods are equal to the same result

$$\begin{aligned} y_1 &= y_0 + \tfrac{3}{8}hf(Y_1) + \tfrac{3}{8}hf(Y_2) + \tfrac{1}{3}hf(Y_3) - \tfrac{1}{12}hf(Y_4) \\ &= y_0 + \tfrac{3}{8}hf(Y_1^*) + \tfrac{3}{8}hf(Y_1^*) + \tfrac{1}{3}hf(Y_2^*) - \tfrac{1}{12}hf(Y_2^*) \\ &= y_0 + \tfrac{3}{4}hf(Y_1^*) + \tfrac{1}{4}hf(Y_2^*). \end{aligned}$$

This is an example of Hundsdorfer–Spijker reducibility [58] (Hundsdorfer, Spijker, 1981).

Experiments with Runge–Kutta methods

The advantages of high order methods

As methods of higher and higher order are used, the cost also increases because the number of f evaluations increases with the number of stages. But using a high order method is usually an advantage over a low order method if sufficient precision is required.

We will illustrate this in Figure 2, where a single half-orbit of the Kepler problem with zero eccentricity is solved using four Runge–Kutta methods ranging from the order 1 Euler method to the methods (1.5 c), (1.5 e) and (1.5 f). The orders of the methods are attached to the plots of their ||error|| versus h behaviours on a log-log scale. Also shown are triangles showing the exact slopes for comparison.

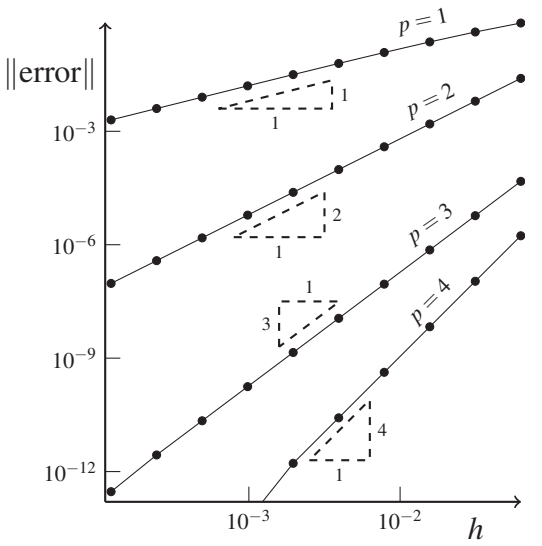


Figure 2 Error behaviour for Runge–Kutta methods with orders $p = 1, 2, 3, 4$, for the Kepler problem with zero eccentricity on the time interval $[0, \pi]$

Methods for stiff problems

The aim in stiff methods is to avoid undue restriction on stepsize for stability reasons but at the same time, to avoid excessive computational cost. In this brief introduction we will compare two methods from the points of view of stepsize restriction, accuracy and cost.

The methods are the third order explicit method (1.5 e) and the implicit Radau IIA method (1.5 h). In each case the problem (1.3 b) (p. 10) was solved with output at $x = 1$ taking n steps with n ranging from 1 to 51200. The dependence of the computational error on n, and therefore on $h = 1/n$ is shown in the Figure 3, where the method used in each result is attached to the curve. Note that the error in the computation is only for a representative component y^1.

From the figure we see that the output for the explicit method is useless unless $h < 0.02$, approximately. This is a direct consequence of the stiffness of the problem. But for the implicit Radau IIA method, there is no constraint on the stepsize except that imposed by the need to obtain sufficient accuracy. Because the computational cost is much greater for the implicit method, many scientists and engineers are willing to use explicit methods in spite of their unstable behaviour and the need to use small stepsizes.

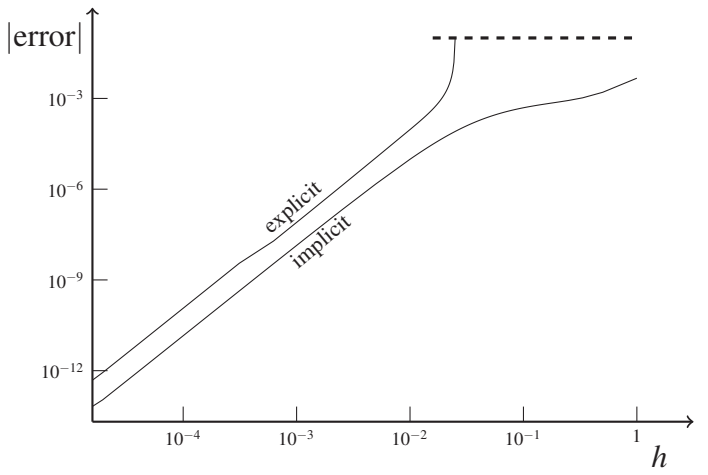


Figure 3 Errors for the stiff problem (1.3 b), solved by an explicit and an implicit method

1.6 Multivalue methods

Linear multistep methods

Instead of calculating a number of stages in working from y_{n-1} to y_n, a linear multistep method makes use of past information evaluated in previous steps. That is, y_n is found from

$$y_n = a_1 y_{n-1} + \cdots + a_k y_{n-k} + hb_1 f(y_{n-1}) + \cdots + hb_k f(y_{n-k}). \tag{1.6 a}$$

In this terminology we will always assume that $|a_k| + |b_k| > 0$ because, if this were not the case, k could be replaced by a lower positive integer. With this understanding, we refer to this as a k-step method. The "explicit case" (1.6 a) is generalized in (1.6 c) below.

In the k-step method (1.6 a), the quantities a_i, b_i, $i = 1, 2, \ldots, k$, are numbers chosen to obtain suitable numerical properties of the method. It is convenient to introduce polynomials ρ, σ defined by

$$\begin{aligned} \rho(w) &= w^k - a_1 w^{k-1} - \cdots - a_k, \\ \sigma(w) &= b_1 w^{k-1} + \cdots + b_k, \end{aligned} \tag{1.6 b}$$

so that the method can be referred to as (ρ, σ) [36] (Dahlquist, 1956).

The class of methods in this formulation can be extended slightly by adding a term $hb_0 f(y_n)$ to the right-hand side of (1.6 a) or, equivalently, a term $b_0 w^k$ to the expression for $\sigma(w)$. Computationally, this means that y_n is defined implicitly as the

solution to the equation

$$y_n - hb_0 f(y_n) = a_1 y_{n-1} + \cdots + a_k y_{n-k} + hb_1 f(y_{n-1}) + \cdots + hb_k f(y_{n-k}).$$

In this case, (1.6 b) is replaced by

$$\begin{aligned} \rho(w) &= w^k - a_1 w^{k-1} - \cdots - a_k, \\ \sigma(w) &= b_0 w^k + b_1 w^{k-1} + \cdots + b_k. \end{aligned} \tag{1.6 c}$$

The most well-known examples of (1.6 b) are the Adams–Bashforth methods [3] (Bashforth, Adams,1883), for which $\rho(w) = w^k - w^{k-1}$ and the coefficients in $\sigma(w)$ are chosen to obtain order $p = k$. Similarly, the well-known Adams–Moulton methods [74] (Moulton, 1926) also have $\rho(w) = w^k - w^{k-1}$ in (1.6 c), but the coefficients in $\sigma(w)$ are chosen to obtain order $p = k+1$.

Consistency, stability and convergence

Definition 1.6A A method (ρ, σ) is preconsistent if $\rho(1) = 0$. The method is consistent if it is preconsistent and also $\rho'(1) = \sigma(1)$.

The significance of Definition 1.6A is that for the problem $y'(x) = 0$, $y(0) = 1$, if $y_{n-i} = 1, i = 1, 2, \ldots, k$, then the value computed by the method in step number n is also equal to the correct value $y_n = 1$ if and only if $\sum_{i=1}^{k} a_i = 1$, which is equivalent to preconsistency. Furthermore, if the method is preconsistent and is used to solve $y'(x) = 1$, $y(0) = 0$, and the values $y_{n-i} = h(n-i)$ then the result computed in step n has the correct value $y_n = nh$ if and only if $nh = \sum_{i=1}^{k} h(n-i)a_i + h\sum_{i=0}^{k} b_i$, which is equivalent to the consistency condition, $k - \sum_{i=1}^{k}(k-i)a_i = \sum_{i=0}^{k} b_i$.

Definition 1.6B A method (ρ, σ) is stable if all solutions of the difference equation

$$y_n = a_1 y_{n-1} + \cdots + a_k y_{n-k}$$

are bounded.

Definition 1.6C A polynomial ρ satisfies the root condition if all zeros are in the closed unit disc and all multiple zeros are in the open unit disc.

The following result follows from the elementary theory of linear difference equations

Theorem 1.6D A method (ρ, σ) is stable if and only if ρ satisfies the root condition.

Exercise 14 Find the values of a_1 and b_1 for which the method $(w^2 - a_1 w + \frac{1}{2}, b_1 w + 1)$ is consistent. Is the resulting method stable?

Order of linear multistep methods

Dahlquist [36] (Dahlquist, 1956) has shown that

> Theorem 1.6E Given $\rho(1) = 0$, the pair (ρ, σ) has order p if and only if
>
> $$\sigma(1+z) = \frac{\rho(1+z)/z}{\ln(1+z)/z} + \mathcal{O}(z^p),$$
>
> where ln denotes the principal value so that $\ln(1+z)/z = 1 + \mathcal{O}(z)$.

For convenience in applications of this result, note that

$$\begin{aligned}\frac{1}{\ln(1+z)/z} = 1 + \frac{1}{2}z - \frac{1}{12}z^2 + \frac{1}{24}z^3 - \frac{19}{720}z^4 + \frac{3}{160}z^5 - \frac{863}{60480}z^6 + \frac{275}{24192}z^7 \\ - \frac{33953}{3628800}z^8 + \frac{8183}{1036800}z^9 + \frac{3663197}{43545600}z^{10} + \mathcal{O}(z^{11}).\end{aligned}$$

Examples of linear multistep methods

The Euler method can be defined by $\rho(w) = w - 1$, $\sigma(w) = 1$ and is the first member of the Adams–Bashforth family of methods [3] (Bashforth, Adams, 1883) The next member is defined by

$$\rho(w) = w^2 - w, \qquad \sigma(w) = \frac{3}{2}w - \frac{1}{2},$$

because

$$\begin{aligned}\sigma(1+z) &= \frac{\rho(1+z)}{z}(1 + \frac{1}{2}z) + \mathcal{O}(z^2) \\ &= (1+z)(1 + \frac{1}{2}z) + \mathcal{O}(z^2) \\ &= 1 + \frac{3}{2}z \\ &= \frac{3}{2}w - \frac{1}{2}, \qquad (w = 1+z)\end{aligned}$$

and has order 2 if correctly implemented. By this is meant the definition of y_1 which is required, in addition to y_0, to enable later values of the sequence of y values to be computed. A simple choice is to define y_1 by a second order Runge–Kutta method, such as (1.5 c) or (1.5 d).

Exercise 15 Show that the order 3 Adams-Bashforth method is defined by $\rho(w) = w^3 - w^2$, $\sigma(w) = \frac{23}{12}w^2 - \frac{4}{3}w + \frac{5}{12}$.

Adams–Moulton methods [74] (Moulton, 1926) are found in a similar way to Adams–Bashforth methods, except that $\sigma(1+z)$ is permitted to have a term in z^k. For $k=2$ and $k=3$, we have in turn $\rho(w)=w-1=z$, $\rho(w)=w^2-w=(1+z)z$, where we will always write $w=1+z$. The formulae for $\sigma(w)$ are, respectively

$$\sigma(w)=1+\tfrac{1}{2}z=\tfrac{1}{2}w+\tfrac{1}{2}, \qquad (k=2),$$
$$\sigma(w)=(1+z)(1+\tfrac{1}{2}z-\tfrac{1}{12}z^2)=1+\tfrac{3}{2}z+\tfrac{5}{12}z^2=\tfrac{5}{12}w^2+\tfrac{2}{3}w-\tfrac{1}{12}, \quad (k=3).$$

Exercise 16 Show that the order 4 Adams-Moulton method is defined by $\rho(w)=w^3-w^2$, $\sigma(w)=\frac{3}{8}w^3+\frac{19}{24}w^2-\frac{5}{24}w+\frac{1}{24}$.

General linear methods

Traditionally, practical numerical methods for differential equations are classified into Runge–Kutta methods and linear multistep methods.

Combining these two families of methods into a single family gives methods characterized by two complexity parameters r, the number of quantities passed from step to step, and s, the number of stages. As for Runge–Kutta methods, the stages will be denoted by $Y_1, Y_2, \ldots, Y_s$ and the corresponding stage derivatives by $F_1, F_2, \ldots, F_s$. The r components of input to step number n will be denoted by $y_1^{[n-1]}, y_2^{[n-1]}, \ldots, y_r^{[n-1]}$, and the output from this step by $y_1^{[n]}, y_2^{[n]}, \ldots, y_r^{[n]}$. These quantities are interrelated in terms of a partitioned $(s+r)\times(s+r)$ matrix

$$\begin{bmatrix} A & U \\ B & V \end{bmatrix}$$

using the equations

$$Y_i = h\sum_{j=1}^{s} a_{ij}F_j + \sum_{j=1}^{r} u_{ij}y_j^{[n-1]}, \quad F_i = f(Y_i), \quad i=1,2,\ldots s,$$
$$y_i^{[n]} = h\sum_{j=1}^{s} b_{ij}F_j + \sum_{j=1}^{r} v_{ij}y_j^{[n-1]}, \qquad i=1,2,\ldots r.$$

The essential part of these relations can be written more compactly as

$$Y = h(A\otimes I)F + (U\otimes I)y^{[n-1]},$$
$$y^{[n]} = h(B\otimes I)F + (V\otimes I)y^{[n-1]},$$

or, if no confusion is possible, as

$$Y = hAF + Uy^{[n-1]},$$
$$y^{[n]} = hBF + Vy^{[n-1]}.$$

Consistency, stability and convergence

Generalizing the ideas of consistency to general linear methods is complicated by the lack of a single obvious interpretation of the information passed between steps of the method. However, we will try to aim for an interpretation in which $y^{[n-1]} = uy(x_{n-1}) + hvy'(x_{n-1}) + \mathcal{O}(h^2)$ for some $u, v \in \mathbb{R}^N$ with the parameters chosen so that at the completion of the step, $y^{[n]} = uy(x_n) + hvy'(x_n) + \mathcal{O}(h^2)$, and also so that the stage values satisfy $Y_i = y(x_{n-1}) + \mathcal{O}(h)$.

We will explore the consequences of these assumptions by analysing the case $n = 1$. We find in turn

$$\begin{aligned} \mathbf{1}y(x_0) &= hAy'(x_0) + U\big(uy(x_0) + hvy'(x_0)\big) + \mathcal{O}(h), \\ U\mathbf{1} &= \mathbf{1}, \\ u(y(x_0) + hy'(x_0) &= hB\big(\mathbf{1}y'(x_0)\big) + V\big(uy(x_0) + hvy'(x_0)\big) + \mathcal{O}(h^2). \end{aligned}$$

For Runge–Kutta methods, there is only a single input and accordingly, $r = 1$. For the method (1.5 f) the defining matrices are

$$\left[\begin{array}{cc} A & U \\ B & V \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 0 & 0 & 0 & 1 \\ \frac{1}{4} & 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & 0 & 0 & 1 \\ 1 & -2 & 2 & 0 & 1 \\ \hline \frac{1}{6} & 0 & \frac{2}{3} & \frac{1}{6} & 1 \end{array}\right].$$

By contrast, for a linear multistep method, $s = 1$. In the case of the order 3 Adams–Bashforth method, the defining matrices are

$$\left[\begin{array}{cc} A & U \\ B & V \end{array}\right] = \left[\begin{array}{c|cccc} 0 & 1 & \frac{23}{12} & -\frac{4}{3} & \frac{5}{12} \\ \hline 0 & 1 & \frac{23}{12} & -\frac{4}{3} & \frac{5}{12} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right].$$

Moving away from traditional methods consider the method with $r = 2$, $s = 3$, with matrices

$$\left[\begin{array}{cc} A & U \\ B & V \end{array}\right] = \left[\begin{array}{ccc|cc} 0 & 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ \hline \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 1 & 0 \\ \frac{1}{4} & -\frac{3}{4} & \frac{1}{2} & 0 & 0 \end{array}\right]. \tag{1.6 d}$$

For a person acquainted only with traditional Runge–Kutta and linear multistep methods, (1.6 d) might seem surprising. However, it is for the analysis of methods like this that the theory of B-series has a natural role. In particular, we note that if the method is started in a suitable manner, then $y_1^{[n]} \approx y(x_n)$ to a similar accuracy as for the fourth order Runge–Kutta method. One possible starting scheme is based on the tableau

$$\mathcal{R}_h = \begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ \hline & -\frac{1}{4} & \frac{1}{8} & \frac{1}{8} & \end{array} .$$

Starting with the initial value y_0, the initial $y^{[0]}$ can be computed by

$$\begin{aligned} y_1^{[0]} &= y_0, \\ y_2^{[0]} &= \mathcal{R}_h y_0 - y_0. \end{aligned} \tag{1.6 e}$$

In Chapter 6, Section 6.4 (p. 225), the method (1.6 d), together with (1.6 e) as starting method, will be used as an illustrative example.

1.7 B-series analysis of numerical methods

Higher derivative methods

The Euler method was introduced in Section 1.4 (p. 14) as the first order case of the Taylor series method. The more sophisticated methods are attempts to improve this basic approximation method.

The practical advantage of methods which require the evaluation of higher derivatives hinges on the relative cost of these evaluations compared with the cost of just the first derivative. But there are other reasons for obtaining formulae for higher derivatives in a systematic way; these are that this information is required for the analysis of so-called B-series.

For a given autonomous problem,

$$y'(x) = f\big(y(x)\big), \quad y(x_0) = y_0, \qquad y : \mathbb{R} \to \mathbb{R}^N, \quad f : \mathbb{R}^N \to \mathbb{R}^N,$$

written in component by component form

$$\frac{\mathrm{d}y^i}{\mathrm{d}x} = f^i(y^1, y^2, \ldots, y^N), \qquad i = 1, 2, \ldots, N,$$

we will find a formula for the second derivative of y^i. This can be obtained by the chain-rule followed by a substitution of the known first derivative of a generic

component f^j. That is,

$$\frac{\mathrm{d}^2 y^i}{\mathrm{d}x^2} = \sum_{j=1}^{N} \frac{\partial f^i}{\partial y^j} \frac{\mathrm{d}y^j}{\mathrm{d}x} = \sum_{j=1}^{N} \frac{\partial f^i}{\partial y^j} f^j.$$

This can be written in a more compact form by using subscripts to indicate partial y derivatives. That is, $f^i_j := \partial f^i / \partial y^j$. A further simplification results by adopting the "summation convention", in which repeated suffixes in expressions like $f^i_j f^j$ imply summation, without this being written explicitly. Hence, we can write

$$\frac{\mathrm{d}^2 y^i}{\mathrm{d}x^2} = f^i_j f^j.$$

Take this further and find formulae for the third and fourth derivatives

$$\frac{\mathrm{d}^3 y^i}{\mathrm{d}x^3} = f^i_{jk} f^j f^k + f^i_j f^j_k f^k,$$
$$\frac{\mathrm{d}^4 y^i}{\mathrm{d}x^4} = f^i_{jk\ell} f^j f^k f^\ell + 3 f^i_{jk} f^j f^k_\ell f^\ell + f^i_j f^j_{k\ell} f^k f^\ell + f^i_j f^j_k f^k_\ell f^\ell.$$

From the sequence of derivatives, evaluated at y_0, the Taylor series can be evaluated.

In further developments, we will avoid the use of partial derivatives, in favour of Fréchet derivatives. That is, in place of the tensors f^i_j, f^i_{jk}, ..., we will use the total derivatives f', f'', ... Evaluated at y_0, these will be denoted by

$$\begin{aligned} \mathbf{f} &= f(y_0), \\ \mathbf{f}' &= f'(y_0), \\ \mathbf{f}'' &= f''(y_0), \\ \vdots &\quad \vdots \end{aligned}$$

Formal Taylor series

The first few terms of the formal Taylor series for the solution at $x = x_0 + h$ are

$$y(x_0 + h) = y_0 + h\mathbf{f} + \tfrac{1}{2}h^2\mathbf{f}'\mathbf{f} + \tfrac{1}{6}h^3\mathbf{f}''\mathbf{f}\mathbf{f} + \tfrac{1}{6}h^3\mathbf{f}'\mathbf{f}'\mathbf{f} + \cdots \tag{1.7 a}$$

Application to the theta method

The result computed by the theta method (1.5 g) (p. 22) has a Taylor expansion, with a resemblance to (1.7 a). That is,

$$y_1 = y_0 + h\mathbf{f} + \theta h^2\mathbf{f}'\mathbf{f} + \tfrac{1}{2}\theta^2 h^3\mathbf{f}''\mathbf{f}\mathbf{f} + \theta^2 h^3\mathbf{f}'\mathbf{f}'\mathbf{f} + \cdots \tag{1.7 b}$$

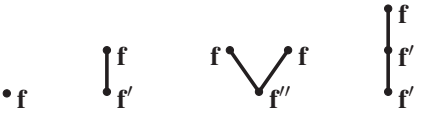
A comparison of (1.7 a) and (1.7 b) suggests that the error in approximating the exact solution by the theta method is $\mathcal{O}(h^2)$ for $\theta \neq \frac{1}{2}$ and $\mathcal{O}(h^3)$ for $\theta = \frac{1}{2}$. Useful though this observation might be, it is just the start of the story. We want to be able to carry out straight-forward analyses of methods using this type of "B-series" expansion. We want to be able to do manipulations of B-series as symbolic counterparts to the computational equations defining the result, and the steps leading to this result, in a wide range of numerical methods.

Elementary differentials and trees

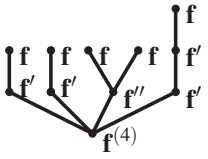
The expressions $\mathbf{f}$, $\mathbf{f}'\mathbf{f}$, $\mathbf{f}''\mathbf{f}\mathbf{f}$ and $\mathbf{f}'\mathbf{f}'\mathbf{f}$ are examples of "elementary differentials" and, symbolically, they have a graph-theoretical analogue. Corresponding to $\mathbf{f}$ is an individual in a genealogical tree; corresponding to $\mathbf{f}'$ is an individual with a link to a possible child. The term $\mathbf{f}'\mathbf{f}$ corresponds to this link having been made to the child represented by $\mathbf{f}$. The bi-linear operator $\mathbf{f}''$ corresponds to an individual with two possible links and in $\mathbf{f}''\mathbf{f}\mathbf{f}$ these links are filled with copies of the child represented by $\mathbf{f}$.

Finally, in these preliminary remarks, $\mathbf{f}'\mathbf{f}'\mathbf{f}$ corresponds to a three generation family with the first $\mathbf{f}'$ playing the role of grandparent, the second $\mathbf{f}'$ playing the role of a parent, and the child of the grandparent; and the final operand $\mathbf{f}$ playing the role of grandchild and child, respectively, of the preceding $\mathbf{f}'$ operators.

The relationship between elementary differentials and trees can be illustrated in diagrams.



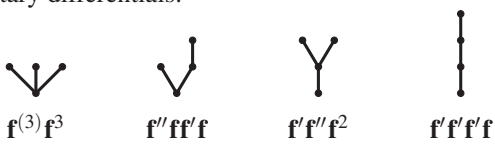
We can extend these ideas to trees and elementary differentials of arbitrary complexity, as shown in the diagram



The elementary differential corresponding to this diagram can be written in a variety of ways. For instance one can insert spaces to emphasize the separation between the four operands of $\mathbf{f}^{(4)}$, or use power notation to indicate repeated operands and operators:

$$\begin{aligned}
&\mathbf{f}^{(4)}\mathbf{f}'\mathbf{f}\mathbf{f}'\mathbf{f}\mathbf{f}''\mathbf{f}\mathbf{f}\mathbf{f}'\mathbf{f}'\mathbf{f} \\
&= \mathbf{f}^{(4)}\ \ \mathbf{f}'\mathbf{f}\ \ \mathbf{f}'\mathbf{f}\ \ \mathbf{f}''\mathbf{f}\mathbf{f}\ \ \mathbf{f}'\mathbf{f}'\mathbf{f} \\
&= \mathbf{f}^{(4)}\ \ (\mathbf{f}'\mathbf{f})^2\ \ \mathbf{f}''\mathbf{f}^2\ \ \mathbf{f}'\mathbf{f}'\mathbf{f} \\
&= \mathbf{f}^{(4)}(\mathbf{f}'\mathbf{f})^2\mathbf{f}''\mathbf{f}^2\mathbf{f}'\mathbf{f}'\mathbf{f}
\end{aligned}$$

As further examples, we show the trees with four vertices, together with the corresponding elementary differentials:



Exercise 17 Find the trees corresponding to each of the elementary differentials:
(a) $\mathbf{f}''(\mathbf{f}'\mathbf{f})^2$, (b) $\mathbf{f}^{(4)}\mathbf{f}^3\mathbf{f}'\mathbf{f}$, (c) $\mathbf{f}'\mathbf{f}'''\mathbf{f}^2\mathbf{f}'\mathbf{f}$.

Exercise 18 Find the elementary differentials corresponding to each of the trees:
(a)  , (b)  , (c)  .

Summary of Chapter 1 and the way forward

Summary

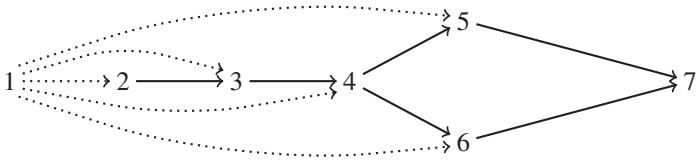
Although this book is focussed on the algebraic analysis of numerical methods, a good background in both ordinary differential equations and numerical methods for their solution is essential.

In this chapter a very basic survey of these important topics has been presented. That is, the fundamental theory of initial value problems is discussed, partly through a range of test problems. These problems arise from standard physical modelling, with the addition of a number of contrived and artificial problems. This is then followed by a brief look at the classical one-step and linear multistep methods, and an even briefer look at some all encompassing multivalue-multistage methods ("general linear methods"). Some of the methods are accompanied by numerical examples, underlining some of their properties.

As a preview for later chapters, B-series are briefly introduced, along with trees and elementary differentials.

The way forward

The current chapter includes preliminary notes on some of the later chapters. This is indicated in the following diagram by a dotted line pointing to these specific chapters. A full line pointing between chapters indicates a stronger prerequisite.



Teaching and study notes

It is a good idea to supplement the reading of this chapter using some of the many books available on this subject. Those best known to the present author are

Ascher, U.M. and Petzold, L.R. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations* (1998) [1]

Butcher, J.C. *Numerical Methods for Ordinary Differential Equations* (2016) [20]

Gear, C.W. *The Numerical Integration of Ordinary Differential Equations* (1967) [44]

Hairer, E., Nørsett, S.P. and Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems* (1993) [50]

Hairer, E. and Wanner G. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems* (1996) [53]

Henrici, P. *Discrete Variable Methods in Ordinary Differential Equations* (1962) [55]

Iserles, A. *A First Course in the Numerical Analysis of Differential Equations* (2008) [61]

Lambert, J.D. *Numerical Methods for Ordinary Differential Systems* (1991) [67]

Projects

Project 1 Explore existence and uniqueness questions for problems satisfying a local Lipschitz condition.

Project 2 Find numerical solutions, using a variety of methods, for the simple pendulum. Some questions to ask are (i) does the quality of the approximations deteriorate with increased initial energy? and (ii) how well preserved is the Hamiltonian.?

Project 3 Learn all you can about fourth order Runge–Kutta methods.

Project 4 Read about predictor-corrector methods in [67] or some other text-book.