UNITEXT 148

Raffaele D'Ambrosio

Numerical Approximation of Ordinary Differential Problems

From Deterministic to Stochastic Numerical Methods

Springer

UNITEXT

La Matematica per il 3+2

Volume 148

The **UNITEXT - La Matematica per il 3+2** series is designed for undergraduate and graduate academic courses, and also includes books addressed to PhD students in mathematics, presented at a sufficiently general and advanced level so that the student or scholar interested in a more specific theme would get the necessary background to explore it.

Originally released in Italian, the series now publishes textbooks in English addressed to students in mathematics worldwide.

Some of the most successful books in the series have evolved through several editions, adapting to the evolution of teaching curricula.

Submissions must include at least 3 sample chapters, a table of contents, and a preface outlining the aims and scope of the book, how the book fits in with the current literature, and which courses the book is suitable for.

For any further information, please contact the Editor at Springer: francesca.bonadei@springer.com

THE SERIES IS INDEXED IN SCOPUS

***

UNITEXT is glad to announce a new series of free webinars and interviews handled by the Board members, who rotate in order to interview top experts in their field.

Access this link to subscribe to the events:
https://cassyni.com/events/TPQ2UgkCbJvvz5QbkcWXo3

Raffaele D'Ambrosio

Numerical Approximation of Ordinary Differential Problems

From Deterministic to Stochastic Numerical Methods

Springer

Raffaele D'Ambrosio
Department of Information Engineering and
Computer Science and Mathematics
University of L'Aquila
L'Aquila, Italy

ISSN 2038-5714 ISSN 2532-3318 (electronic)
UNITEXT
ISSN 2038-5722 ISSN 2038-5757 (electronic)
La Matematica per il 3+2
ISBN 978-3-031-31342-4 ISBN 978-3-031-31343-1 (eBook)
https://doi.org/10.1007/978-3-031-31343-1

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

To Raffaela, my grandmother, whose love for me has never passed away

Preface

This book is devoted to the numerical discretization of ordinary differential equations (ODEs), here presented under several perspectives. First of all, the attention is conveyed to the basic aspects of the numerical approximation of ODEs, with clear emphasis on providing accurate numerical solutions of deterministic problems. Then, the focus is moved to a more modern vision of numerical approximation, oriented to reproducing qualitative properties of the continuous problem along the discretized dynamics. Indeed, modern Numerical Analysis is not only devoted to accurately approximating the solutions of various problems through efficient and robust schemes, but also to retaining the characteristic properties of the continuous problem over long times. Sometimes such conservation properties naturally characterize the numerical schemes, while in more complex situations preservation issues have to be conveyed into the numerical approximations.

The book also performs some steps in the direction of stochastic differential equations (SDEs), with the intention of offering useful tools to generalize the techniques introduced for the numerical approximation of ODEs to the stochastic case, as well as of presenting those natively introduced for SDEs.

The book represents the result of an intense teaching experience as well as of the research carried out in the last decade by the author. It is both intended for students and instructors: for the students, this book is comprehensive and mostly self-contained; for the instructors, there is material for one (or even more than one) monographic course on ODEs and related topics. In this respect, the book can be followed in its designed path and includes motivational aspects, historical background and examples. All theoretical issues are thought for a better understanding of the many aspects of numerical approximation, even when the theorems are mostly dealing with the differential problem rather than with its numerical counterpart. The author is indeed convinced that understanding as much as possible on the problem is a key aspect in designing proper numerical solvers. The book is also equipped with a large number of software programs, implemented in MATLAB, that can be useful for the laboratory part of a course in numerical ODEs.

There are several beautiful monographs on the numerical approximations of the solutions to ODEs, so a natural question is: "Why yet another book?". Actually, the effort of the author in writing this monograph was aimed at

- regarding the theoretical analysis of ODEs as a building block for the numerical approximation, inferring relevant qualitative features of the continuous problem to be imitated along the discretized dynamics, in a way that removes possible useless dichotomies between the study of the problem and that of its approximation;
- presenting the well-established theory of numerical methods for ODEs in a comprehensive way, with elements of novelty in the analysis that provide a simplified framework without sacrificing the rigor of the presentation;
- covering the case of SDEs, which have played a relevant role in the recent literature on numerical analysis of evolutive problems and have not yet been included in a comprehensive monograph on ODEs;
- including a wide practical setting, not only given by the provided software, but also (and especially) by the analysis of the results that assess the effectiveness of the presented approaches and confirm the theoretical results.

Together with strictly mathematical aspects, the book contains the portraits of several pioneers in the numerical discretization of ODEs.

My most sincere gratitude goes to Alfio Quarteroni, who encouraged me a lot to write this book and gave me many precious suggestions useful to improve the presentation of the monograph. I am thankful to Beatrice Paternoster, who guided me along the path leading to my professorship with great, sincere care. I am thankful to Zdzislaw Jackiewicz, especially for the great time we spent doing research together during my visit to Arizona when I was a Ph.D. student. I am grateful to John Butcher, who inspired me a lot with his deep love for Numerical Analysis and his heartfelt attitude; we have spent a lot of time together all over the world as well as during my visits to New Zealand, where we had several occasions to enjoy Mathematics and... singing together. I am beholden to Ernst Hairer, from whom I learned a lot on geometric numerical integration, especially during my visit to Geneva, which was highly inspiring for me. I am thankful to Luca Dieci for his warm hospitality while I was Fulbright Research Scholar at Georgia Institute of Technology, where I have learned most of what I know on piecewise smooth dynamical systems. I am grateful to Evelyn Buckwar for introducing me to the world of numerical integration for SDEs during my visit to her in Linz. Many thanks to Kevin Burrage, David Cohen, Hugo de la Cruz and Gilles Vilmart for all profitable discussions on the numerics for SDEs. Thanks a lot to Springer Nature, Italy, especially to Francesca Bonadei for her precious support and encouragement and to the anonymous referee for carefully reading the manuscript.

There are many colleagues, students and friends to whom I am deeply grateful, but a detailed list would certainly leave someone out: whoever you are, you will recognize yourself in my deepest gratitude.

Last but not least, thanks a lot to you, reader of this book. I hope you will enjoy it and share my love for the topic, which I sincerely hope will emerge from each page.

L'Aquila, Italy Raffaele D'Ambrosio
January 2023

Contents

Chapter 1
Ordinary Differential Equations

In order to solve this differential equation you look at it until a solution occurs to you.

(George Polya, How to Solve It: A New Aspect of Mathematical Method, Princeton University Press, 1945)

In order to properly address the issue of numerically solving any mathematical problem, it is always extremely important to understand as much as possible the problem itself. Hence, this chapter focuses on some basic issues on initial value problems for ordinary differential equations, and in particular on the well-posedness of the problem and the stability of solutions. Of course, this chapter does not pursue the aim of being a comprehensive treatise on the theory of ODEs; rather, the results here presented are clearly meant to provide significant issues relevant for computational purposes. The reader interested in monographs specifically dedicated to presenting a general theory of ODEs under a qualitative point of view can refer, for instance, to [14, 24, 90, 99, 199–201].

1.1 Initial Value Problems

Consider the following initial value problem for first order ordinary differential equations (ODEs)

$$\begin{cases} y'(t) = f(t, y(t)), \\ y(t_0) = y_0, \end{cases} \tag{1.1}$$

with $t \in [t_0, T]$, $f : [t_0, T] \times \mathbb{R}^d \to \mathbb{R}^d$ and $y_0 \in \mathbb{R}^d$. The right-hand side of the differential equation in (1.1), also denoted as *vector field* of the problem, can be of the form $f(y(t))$, i.e., dependent on t only through the solution $y(t)$: in this case the problem is said to be *autonomous*. When the vector field explicitly depends on the independent variable t, as in (1.1), the problem is said to be *non-autonomous*. In the

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_1

remainder of the treatise, we will always refer to the non-autonomous form (1.1), unless explicitly specified.

Problem (1.1) can be assumed as a prototype model of evolution in time and, indeed, it models many real life phenomena. Moreover, it also arises in the spatial discretization of time-dependent partial differential equations: therefore, it assumes an important and general role in mathematical modeling of evolutive problems. As a consequence, understanding the problem in depth, as well as giving proper methodologies for its solution, deserves the necessary attention. Before entering into both topics, let us now provide some examples of models described by ODEs, which have a specific meaning in applications.

Example 1.1 (Dynamics of Social Networks) Social networks have an intrinsic dynamical behavior with rapid evolution in time, due to the large amount of real-time interactions among their users. It looks particularly interesting to study the dynamics of processes over the network as well as the underlying connectivity among users, as a prototype model of human interactions.

Let us suppose that N is the number of users of a selected social network and denote by $A(t) \in \mathbb{R}^{N\times N}$ the corresponding binary time-dependent adjacency matrix, i.e.,

$$A_{ij}(t) = \begin{cases} 1, & \text{if and only if } i \text{ communicates with } j \text{ at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

We observe that $A(t)$ is a symmetric matrix in case, for instance, of voice calls and a generic entry differs from 0 only when the call is active. In other terms, $A(t)$ measures the effective interaction among two users, clearly clarifying also the time when such an interaction starts and when it stops.

The non-symmetric and non-binary case occurs, for instance, in case of e-mails and tweets and, in this case, a vanishing value of an element of $A(t)$ highlights a possible delay in receiving the message or sending a reply (in the case of e-mails) or the loss of visibility of a message over time (in the case of tweets).

Mantzaris and Higham introduced in [258] the notion of *dynamic communicators* to denote individuals able to disseminate or collect information. In [179], Grindrod and Higham provided an elegant model to describe the evolution of dynamic communicability by means of a matrix differential equation in the unknown $S(t) \in \mathbb{R}^{N\times N}$, which is a real-valued non-symmetric matrix whose general entry $S_{ij}(t)$ describes the ability of an individual i to communicate with another individual j at time t. Such a model is given by

(continued)

Example 1.1 (continued)
the following matrix differential equation

$$U'(t) = -b(U(t) - I) - U(t)\log(I - aA(t)), \tag{1.2}$$

where $U(t) = I + S(t)$, I is the identity matrix in $\mathbb{R}^{N\times N}$, a, b are positive parameters. Equation (1.2) requires the computation of a matrix logarithm; functions of matrices are extensively described in [211]. Here we briefly highlight that, for a stable matrix M (i.e., whose spectral radius is smaller than 1), the matrix $\log(I+M)$ is the sum of Mercator power series $\sum_{k=1}^{\infty} \frac{(-1)^k}{k} M^k$.

Example 1.2 (Hodgkin-Huxley Model, a Nobel Prize System of ODEs) In 1963, the Nobel Prize in Physiology or Medicine was assigned to Sir Alan Lloyd Hodgkin and Sir Andrew Fielding Huxley, two English physiologists and biophysicists who gave in [216] an accurate description of excitation and inhibition mechanisms in the cell membrane, published after a series of more qualitative contributions on the flow of electric current through the layer membrane of a nerve fibre. Their model obeys the following nonlinear system of ODEs, describing the dynamics of the functions $V(t)$, $n(t)$, $m(t)$, $h(t)$ (for a more amenable notation, time dependency is omitted in the remainder):

$$\begin{aligned}
I &= C_m V' + \overline{g}_K n^4 (V - V_K) + \overline{g}_{Na} m^3 h (V - V_{Na}) + \overline{g}_l (V - V_l),\\
n' &= \alpha_n(V)(1 - n) - \beta_n(V) n,\\
m' &= \alpha_m(V)(1 - m) - \beta_m(V) m,\\
h' &= \alpha_h(V)(1 - h) - \beta_h(V) h,
\end{aligned}$$

where I is the total membrane current, C_m is the membrane capacity per unit area (i.e., the ability to store an electric charge), V is the membrane potential, $\overline{g}_K$ is a constant proportional to potassium conductance g_K (indeed, $g_K = \overline{g}_K n^4$ or, in other terms, potassium ions can only cross the membrane when four similar particles occupy a certain region of the membrane), n is the proportion of particles inside the membrane (consequently, $1 - n$ is the proportion of particles outside of the membrane), m provides the proportion of activating molecules inside the membrane, h is the proportion of inactivating molecules outside the membrane, $\overline{g}_l$ is a constant proportional to leakage conductance. The values of V_K, V_{Na} and V_l are reversal potentials of sodium, potassium and of the leakage current, respectively. Transfer rates from outside

(continued)

Example 1.2 (continued)
to inside and vice versa are respectively given by the functions

$$\alpha_n(V) = 0.01(V+10)\left(\mathrm{e}^{\frac{10-V}{10}} - 1\right)^{-1}, \quad \beta_n(V) = 0.125\mathrm{e}^{-\frac{V}{80}}.$$

Transfer rates of activating molecules from inside to outside and vice versa are modeled by

$$\alpha_m(V) = 0.01(V+25)\left(\mathrm{e}^{\frac{25-V}{10}} - 1\right)^{-1}, \quad \beta_m(V) = 4\mathrm{e}^{-\frac{V}{18}}.$$

Finally, the rate of transfer of inactivating molecules from outside to inside are

$$\alpha_h(V) = 0.07\mathrm{e}^{-\frac{V}{20}}, \quad \beta_h(V) = \left(\mathrm{e}^{\frac{30-V}{10}} + 1\right)^{-1}.$$

Example 1.3 (Homeostasis of T-cells) Homeostasis can be defined as the natural property of living organisms to preserve their internal stability, in response to changes in external conditions. A specific example of homeostasis regards the immune system, where the number of cells playing a key role in the immune response (the so-called T-cells) is normally retained along the adult life of an individual [76]. Homeostasis of T-cells is believed to be due to a *quorum-sensing* mechanism, normally typical of bacteria, according to which CD4+ cells (i.e., the T-cells responsible of activating the immune response, by sending signals to another family of T-cells, the so-called CD8 killer cells) could control their own expansion, thanks to their ability to perceive the density of their own populations, in order to prevent uncontrolled lymphocyte proliferation during immune responses.

In this homeostatic mechanism Interleukin-2 (IL-2), which is a protein regulating the activities of lymphocytes responsible for immunity, plays a significant role, very well described for instance in [10] and references therein. In [10], the authors have developed an ODE-based model describing CD4+ T-cell homeostasis in terms of the time evolution of the following cellular populations: $n_1(t)$, describing naïve T-cells; $n_2(t)$, characterizing IL-2 producing cells; $n_3(t)$, denoting activated/memory non-IL-2 producing cells; $n_4(t)$, modeling regulatory CD4+ T-cells.

(continued)

Example 1.3 (continued)
The corresponding system of ODEs assumes the following form [10]:

$$1000\, n_1'(t) = -n_1(t)\left(91 + \frac{n_1(t)}{50}\right),$$

$$n_2'(t) = 2\left(500 n_1(t) + 5 n_3(t) + \frac{3 n_2^2(t)}{20} - n_2(t) n_4(t)\right),$$

$$100\, n_3'(t) = n_1(t) + n_2(t) + n_3(t)\left(\frac{59}{10} + \frac{n_2(t)}{500} - \frac{n_3(t)}{200}\right) + \frac{n_4(t)}{50},$$

$$100\, n_4'(t) = -n_4(t)\left(\frac{10}{10 + n_2(t)} - \frac{n_2(t)}{100}\right),$$

with initial conditions

$$n_1(0) = 100, \quad n_2(0) = n_3(0) = 0, \quad n_4(0) = 1.$$

As claimed by the authors in [10], the quorum-sensing mechanism described by above equations provides that regulatory T-cells count and regulate the number of activated T-cells through the detection of IL-2 and the number of interactions between these two populations, whose specified proportion (encoded within the parameters of the model) leads to cellular events such as division, survival or suppression.

Example 1.4 (Fake News Diffusing as Epidemics) Internet is certainly a primary medium of rapidly accessible information. Clearly, since veracity is a serious issue in the spread out of online information, especially in social networks, the circulation of fake news has been extensively studied during last years. Understanding mechanisms of fake news diffusion and, hopefully, predicting their growth in order to favor the reaffirmation of the truth, is a very important task in the age of digitalization.

A widespread point of view in the existing literature conceives the diffusion of fake news similar to that of an epidemic (see, for instance, [137, 166, 257, 278] and references therein). In this direction, an ODE-based model for the circulation of fake information has been presented in [137], together with its stiffness analysis (the concept of stiff problems will be introduced in Chap. 7 of this book) useful to understand how fast is the transit

(continued)

Example 1.4 (continued)
of fake information in a given country. Modeling aspects in [137] are based on the well known SIR model, widely studied by the existing literature regarding mathematical epidemiology [28–30, 77, 154, 163, 236]. SIR model describes the effects of the diffusion of an epidemic in a population ideally divided into three groups of individuals (susceptible, infected and recovered people) that, in the context of fake news, can be interpreted as follows:

- $S(t)$, the population of potential authors of fake news;
- $I(t)$, collecting all active authors in posting fake information;
- $R(t)$, grouping inactive fake news authors (for instance, recovered after fact checking).

The mutual interactions among these three populations are described by the following system of ODEs:

$$\begin{cases} S'(t) = -\beta S(t) I(t), \\ I'(t) = \beta S(t) I(t) - \alpha I(t), \\ R'(t) = \alpha I(t), \end{cases} \tag{1.3}$$

where α and β are recovery and contact rates, respectively. In [137], the values of these parameters have been related with human development and internet penetration indices h and i, provided in the annual report of United Nations Development Programme for all countries. Specifically,

$$\alpha = \frac{h}{100}, \quad \beta = \frac{i}{10},$$

and, in general, the value of α is smaller than that of β, since spreading a lie is much easier than reaffirming the truth. Moreover, as highlighted in [327], truth reaffirmation is never viral and requires a large human commitment and this is the only option to restore the truth. On the contrary, the spread of fake information does not necessarily require a strong human presence, since authors are frequently bots or fake accounts. As a consequence, it is worth relating the recovery rate α in a given country to its human development index and the contact rate β to its internet penetration index.

1.2 Well-Posedness

A key issue in the context of initial value problems (1.1), which also plays a central role in their numerical approximation, is given by well-posedness. The notion of well-posedness we adopt is that of Hadamard: problem (1.1) is said *well-posed* if

- a solution exists;
- such a solution is unique;
- the solution continuously depends on problem data.

Let us analyze each condition in details, starting with the existence of a solution to (1.1). To this purpose, let us show a fundamental existence result due to Peano, who provided a first proof in 1886 [284], later improved in 1890 [285], although the mostly given one relies on Arzelà-Ascoli theorem, here briefly recalled (a complete proof can be found, for instance, in [302]).

Theorem 1.1 (Arzelà-Ascoli Theorem) *Let $\{y_n(t)\}_{n\in\mathbb{N}}$, with $y_n : [t_0, T] \to \mathbb{R}^d$, be a sequence of functions satisfying the following properties:*

- *equicontinuity, i.e., for any $\varepsilon > 0$ there exists $\delta > 0$, such that $|t_2 - t_1| < \delta$ implies $\|y_n(t_2) - y_n(t_1)\| < \varepsilon$, for any $n \in \mathbb{N}$. Here $\|\cdot\|$ denotes a suitable norm of $\mathbb{R}^d$;*
- *uniform boundedness, i.e., there exists $M > 0$, such that $\|y_n(t)\| < M$, for any $t \in [t_0, T]$.*

Then, the sequence $\{y_n(t)\}_{n\in\mathbb{N}}$ contains a subsequence that is uniformly convergent in $[t_0, T]$.

Theorem 1.2 (Peano Theorem) *Let $f : [a, b] \times D \to \mathbb{R}^d$, with $D \subseteq \mathbb{R}^d$, be a continuous function. Consider $(t_0, y_0) \in [a, b] \times D$ and $\delta, \tau > 0$, such that $\mathcal{S} \subseteq [a, b] \times D$, having denoted $\mathcal{S} = [t_0 - \tau, t_0 + \tau] \times \mathcal{B}_\delta(y_0)$ and being $\mathcal{B}_\delta(y_0)$ the ball centered in y_0 with radius δ.*

Then, the initial value problem (1.1) has at least one solution $y(t)$, for $t \in [t_0 - \gamma, t_0 + \gamma]$, with

$$\gamma \leq \min\left\{\tau, \frac{\delta}{M}\right\},$$

being

$$M = \max_{(t,y)\in\mathcal{S}} |f(t, y(t))|. \tag{1.4}$$

Proof The idea of the proof consists in constructing a sequence of functions fulfilling the hypothesis of Arzelà-Ascoli theorem, i.e., an equicontinuous and uniformly bounded sequence of functions, from which we aim to extract a subsequence converging to the requested solution of problem (1.1). The proof is given when $t \in [t_0, t_0 + \gamma]$ and, *mutatis mutandis*, it can be given for $t \in [t_0 - \gamma, t_0]$ in similar way.

Problem (1.1) admits the following equivalent integral formulation

$$y(t) = y_0 + \int_{t_0}^{t} f(s, y(s))\mathrm{d}s$$

and, as announced, we consider $t \in [t_0, t_0+\gamma]$. Let us define the following sequence $\{y_n(t)\}_{n\in\mathbb{N}^+}$ of continuous functions in $[t_0, t_0 + \gamma]$ by

$$y_n(t) = \begin{cases} y_0, & t \in \left[t_0, t_0 + \dfrac{\gamma}{n}\right], \\ y_0 + \displaystyle\int_{t_0}^{t-\frac{\gamma}{n}} f(s, y_n(s))\mathrm{d}s, & t \in \left(t_0 + \dfrac{\gamma}{n}, t_0 + \gamma\right]. \end{cases} \tag{1.5}$$

Let us analyze some properties of this sequence:

- first of all, $y_n(t) \in \mathcal{B}_\delta(y_0)$, for any $t \in [t_0, t_0 + \gamma]$, since

$$\|y_n(t) - y_0\| \le \left\| \int_{t_0}^{t-\frac{\gamma}{n}} f(s, y_n(s))\mathrm{d}s \right\| \le \int_{t_0}^{t-\frac{\gamma}{n}} \|f(s, y_n(s))\|\mathrm{d}s$$

$$\le M(t - t_0) \le M\gamma \le \delta;$$

- the sequence with general term given by (1.5) is uniformly bounded. Indeed,

$$\|y_n(t)\| \le \left\| y_0 + \int_{t_0}^{t-\frac{\gamma}{n}} f(s, y_n(s))\mathrm{d}s \right\| \le \|y_0\| + \int_{t_0}^{t-\frac{\gamma}{n}} \|f(s, y_n(s))\|\,\mathrm{d}s$$

$$\le \|y_0\| + \delta;$$

- finally, such a sequence is equicontinuous. Indeed, for an arbitrary $\varepsilon > 0$, we prove the existence of $\sigma > 0$ leading to $\|y_n(t_2) - y_n(t_1)\| < \varepsilon$, for any couple of values $t_1, t_2 \in [t_0, t_0 + \gamma]$ such that $|t_2 - t_1| < \sigma$.

We have

$$\|y_n(t_2) - y_n(t_1)\| \leq \left\| y_0 + \int_{t_0}^{t_2 - \frac{\gamma}{n}} f(s, y_n(s))\mathrm{d}s - y_0 - \int_{t_0}^{t_1 - \frac{\gamma}{n}} f(s, y_n(s))\mathrm{d}s \right\|$$

$$\leq \int_{t_1 - \frac{\gamma}{n}}^{t_2 - \frac{\gamma}{n}} \|f(s, y_n(s))\| \,\mathrm{d}s \leq M|t_2 - t_1|.$$

Then, it is enough to choose $\sigma \leq \varepsilon/M$, in order to get

$$\|y_n(t_2) - y_n(t_1)\| \leq \varepsilon.$$

As a consequence of the aforementioned properties, by means of Arzelà-Ascoli theorem, there exists a subsequence $\{y_{n_j}(t)\}_{j\in\mathbb{N}^+}$ of $\{y_n(t)\}_{n\in\mathbb{N}^+}$ uniformly converging to a function $\overline{y}(t)$. Since uniform convergence retains continuity, function $\overline{y}(t)$ is continuous.

Let us write the system corresponding to (1.5) satisfied by $\{y_{n_j}(t)\}_{j\in\mathbb{N}^+}$, i.e.,

$$y_{n_j}(t) = \begin{cases} y_0, & t \in \left[t_0, t_0 + \dfrac{\gamma}{n_j}\right], \\ y_0 + \displaystyle\int_{t_0}^{t - \frac{\gamma}{n_j}} f(s, y_{n_j}(s))\mathrm{d}s, & t \in \left(t_0 + \dfrac{\gamma}{n_j}, t_0 + \gamma\right] \end{cases}$$

and let us recast last equation as

$$y_{n_j}(t) = y_0 + \int_{t_0}^{t} f(s, y_{n_j}(s))\mathrm{d}s - \int_{t - \frac{\gamma}{n_j}}^{t} f(s, y_{n_j}(s))\mathrm{d}s.$$

In the limit when j tends to infinity,

- $y_{n_j}(t)$ tends to $\overline{y}(t)$, by Arzelà-Ascoli theorem;
- $\displaystyle\int_{t_0}^{t} f(s, y_{n_j}(s))\mathrm{d}s$ tends to $\displaystyle\int_{t_0}^{t} f(s, \overline{y}(s))\mathrm{d}s$, by the continuity of the function f;
- $\displaystyle\int_{t - \frac{\gamma}{n_j}}^{t} f(s, y_{n_j}(s))\mathrm{d}s$ tends to 0, since

$$\left\| \int_{t - \frac{\gamma}{n_j}}^{t} f(s, y_{n_j}(s))\mathrm{d}s \right\| \leq \int_{t - \frac{\gamma}{n_j}}^{t} \|f(s, y_{n_j}(s))\|\mathrm{d}s \leq M\frac{\gamma}{n_j}$$

and $M\frac{\gamma}{n_j}$ tends to 0 when j tends to infinity.

Therefore,

$$\overline{y}(t) = y_0 + \int_{t_0}^{t} f(s, \overline{y}(s))\mathrm{d}s, \quad t \in [t_0, t_0 + \gamma].$$

□

Clearly, Peano theorem only provides the existence of the solution of (1.1), under the assumption of continuity of the vector field. We now aim to understand under which assumptions the solution is also unique, through a result constructing the solution itself. To this purpose, we need the following preliminary lemma.

Lemma 1.1 (Right Grönwall Lemma) *Let $\alpha(t), y(t) : [t_0, \infty) \to \mathbb{R}$ be continuous functions, with $\alpha(t) \geq 0$, for any $t \geq t_0$. If there exists $M \in \mathbb{R}$, such that*

$$y(t) \leq M + \int_{t_0}^{t} \alpha(s)y(s)\mathrm{d}s, \quad t \geq t_0, \tag{1.6}$$

then,

$$y(t) \leq M \exp\left(\int_{t_0}^{t} \alpha(s)\mathrm{d}s\right), \quad t \geq t_0. \tag{1.7}$$

Proof Let us define the auxiliary function

$$z(t) = M + \int_{t_0}^{t} \alpha(s)y(s)\mathrm{d}s,$$

describing the right-hand side of (1.6). According to the fundamental theorem of calculus,

$$z'(t) = \alpha(t)y(t)$$

and, by (1.6), we obtain

$$z'(t) - \alpha(t)z(t) \leq 0.$$

We multiply each side of the last inequality by $\exp\left(-\int_{t_0}^{t} \alpha(s)\mathrm{d}s\right)$, leading to

$$\left(z'(t) - \alpha(t)z(t)\right) \exp\left(-\int_{t_0}^{t} \alpha(s)\mathrm{d}s\right) \leq 0.$$

Taking into account that the left-hand side of this inequality is the first derivative of $z(t)\exp\left(-\int_{t_0}^{t}\alpha(s)\mathrm{d}s\right)$, this function is certainly non-increasing. Then,

$$z(t)\exp\left(-\int_{t_0}^{t}\alpha(s)\mathrm{d}s\right)\leq z(t_0)\exp\left(-\int_{t_0}^{t_0}\alpha(s)\mathrm{d}s\right)=M$$

or, equivalently,

$$z(t)\leq M\exp\left(\int_{t_0}^{t}\alpha(s)\mathrm{d}s\right).$$

Combining last equation with (1.6) gives the thesis. □

This result was proved in 1919 by the Swedish mathematician Thomas Hakon Grönwall (1877–1932) [180]. It is a very useful tool anytime an implicit estimate of a certain function, such as that in (1.6), has to be made explicit, as in (1.7). Similarly, as visible from the proof, Grönwall lemma is also relevant in giving an estimate of a function for which a differential inequality is known. Finally, we also notice that Theorem 1.1 admits several generalizations, leading to proper fully explicit Grönwall-type estimates associated to implicit inequalities involving a certain function. For instance, in analogous way, one can prove a left version of Lemma 1.1, which is here stated, leaving the similar proof to the reader.

Lemma 1.2 (Left Grönwall Lemma) *Let $\alpha(t), y(t):(-\infty,t_0]\to\mathbb{R}$ be continuous functions, with $\alpha(t)\geq 0$, for any $t\leq t_0$. If there exists $M\in\mathbb{R}$ such that*

$$y(t)\geq M+\int_{t}^{t_0}\alpha(s)y(s)\mathrm{d}s,\quad t\leq t_0,$$

then,

$$y(t)\geq M\exp\left(\int_{t}^{t_0}\alpha(s)\mathrm{d}s\right),\quad t\leq t_0.$$

We are now able to prove the following existence and uniqueness theorem, well known as Picard-Lindelöf theorem, or Cauchy-Lipschitz theorem. The contributions by Charles Èmile Picard (1856–1941) and Ernst Leonard Lindelöf (1870–1946) were respectively published in 1890 [288] and 1894 [252], therefore after those by Peano. As it arises from the proof, this result has a twofold value: on the one hand, it gives conditions ensuring the existence and the uniqueness of the solution

to (1.1); on the other one, it suggest a preliminary numerical scheme for the solution of (1.1).

Theorem 1.3 (Picard-Lindelöf Theorem) *Let $f : [a, b] \times D \to \mathbb{R}^d$, where $D \subseteq \mathbb{R}^d$, be a continuous and Lipschitz continuous function with respect to its second argument, i.e., there exists $L \geq 0$ such that*

$$\|f(t, y_1(t)) - f(t, y_2(t))\| \leq L \|y_1(t) - y_2(t)\|, \tag{1.8}$$

for any $(t, y_1(t)), (t, y_2(t)) \in [a, b] \times D$, being $\|\cdot\|$ any norm in $\mathbb{R}^d$. Consider $(t_0, y_0) \in [a, b] \times D$ and $\delta > 0$, $\tau > 0$, such that

$$\mathcal{S} \subseteq [a, b] \times D,$$

where $\mathcal{S}$ is the set defined in the statement of Theorem 1.2. If $\gamma \leq \min\{\tau, \frac{\delta}{M}\}$, with M defined as in (1.4), then the initial value problem (1.1) has a unique solution $y(t)$, for $t \in [t_0 - \gamma, t_0 + \gamma]$.

Proof The proof is given for $t \in [t_0, t_0 + \tau]$; the case $t \in [t_0 - \tau, t_0]$ can be analogously proved. We first prove the uniqueness of the solution, by contradiction. Indeed, we suppose that two solutions $y(t)$ and $w(t)$ of (1.1) exist. In particular, they also satisfy (1.1) in its integral form, i.e.

$$y(t) = y_0 + \int_{t_0}^{t} f(s, y(s))\mathrm{d}s,$$

$$w(t) = y_0 + \int_{t_0}^{t} f(s, w(s))\mathrm{d}s.$$

Side-by-side subtracting and passing to the norm yields

$$\|y(t) - w(t)\| \leq \int_{t_0}^{t} \|f(s, y(s)) - f(s, w(s))\|\mathrm{d}s.$$

Moreover, Lipschitz continuity of f gives

$$\|y(t) - w(t)\| \leq L \int_{t_0}^{t} \|y(s) - w(s)\|\mathrm{d}s$$

and, by the right Grönwall lemma 1.1 with $M = 0$ and $\alpha(s) = L$, we obtain

$$\|y(t) - w(t)\| \leq 0,$$

i.e., $y(t) = w(t)$, for $t \in [t_0, t_0 + \tau]$. We also observe that, by the left Grönwall lemma 1.2, we have $y(t) = w(t)$, also for $t \in [t_0 - \tau, t_0]$, hence

$$y(t) = w(t), \qquad t \in [t_0 - \tau, t_0 + \tau],$$

giving the desired uniqueness in the whole interval $t \in [t_0 - \tau, t_0 + \tau]$.

Though existence has already been proved in Peano theorem 1.2, we provide existence proof by Picard and Lindelöf completely from scratch, since it leads to useful numerical considerations. To this purpose, we introduce the following recursively defined sequence

$$\begin{aligned} y_0(t) &= y_0, \\ y_{n+1}(t) &= y_0 + \int_{t_0}^{t} f(s, y_n(s))\mathrm{d}s, \qquad n \geq 0. \end{aligned} \tag{1.9}$$

First of all, we aim to prove that it is well-defined, i.e., that for any $t \in [t_0, t_0 + \tau]$, $y_n(t)$ belongs to $\mathcal{B}_\delta(y_0)$. Let us proceed by induction. If $n = 0$, $y_0(t) = y_0$ trivially belongs to $\mathcal{B}_\delta(y_0)$, for any $t \in [t_0 - \tau, t_0 + \tau]$. Now, let us suppose that $y_n(t) \in \mathcal{B}_\delta(y_0)$ and restrict our attention to $t \in [t_0, t_0 + \tau]$.

We have

$$\begin{aligned} \|y_{n+1}(t) - y_0\| &= \left\| \int_{t_0}^{t} f(s, y_n(s))\mathrm{d}s \right\| \leq \int_{t_0}^{t} \|f(s, y_n(s))\|\mathrm{d}s \\ &\leq M(t - t_0) \leq M\gamma \leq \delta. \end{aligned}$$

We now prove, by induction, the following bound:

$$\|y_{n+1}(t) - y_n(t)\| \leq \frac{ML^n(t - t_0)^{n+1}}{(n+1)!}, \quad n \geq 0. \tag{1.10}$$

The case $n = 0$ is trivial, since

$$\|y_1(t) - y_0(t)\| = \left\| \int_{t_0}^{t} f(s, y_0(s))\mathrm{d}s \right\| \leq \int_{t_0}^{t} \|f(s, y_0(s))\|\mathrm{d}s \leq M(t - t_0).$$

Since

$$\|y_{n+1}(t) - y_n(t)\| \leq \int_{t_0}^{t} \|f(s, y_n(s)) - f(s, y_{n-1}(s))\|\,\mathrm{d}s,$$

by the Lipschitz continuity of f and using the inductive hypothesis, we obtain

$$\|y_{n+1}(t) - y_n(t)\| \leq \frac{ML^n}{n!} \int_{t_0}^{t} (s - t_0)^n \mathrm{d}s = \frac{ML^n(t - t_0)^{n+1}}{(n+1)!}.$$

The term

$$\frac{L^{n+1}(t-t_0)^{n+1}}{(n+1)!}$$

is the leading term of a power series uniformly converging to $e^{L(t-t_0)}$ in $[t_0, t_0+\tau]$. As a consequence, the series

$$y_0(t) + \sum_{n=0}^{\infty} (y_{n+1}(t) - y_n(t))$$

is uniformly convergent in $[t_0, t_0+\tau]$. We observe that

$$y_k(t) = y_0 + \sum_{n=0}^{k-1} (y_{n+1}(t) - y_n(t))$$

and, therefore, the sequence $\{y_k(t)\}_{k\in\mathbb{N}}$ is also uniformly convergent in $[t_0, t_0+\tau]$ and we denote its limit by $z(t)$.

With similar arguments, we can prove that the sequence of integrals

$$\left\{ \int_{t_0}^{t} f\,(s, y_n(s))\,\mathrm{d}s \right\}_{n\in\mathbb{N}}$$

is uniformly convergent in $[t_0, t_0+\tau]$ to $\int_{t_0}^{t} f(s, z(s))\mathrm{d}s$. In summary, when n tends to infinity, we get from (1.9) that

$$z(t) = y_0 + \int_{t_0}^{t} f(s, z(s))\mathrm{d}s,$$

that concludes the proof of existence when $t \in [t_0, t_0+\tau]$. As aforementioned, an analogous proof of existence can be given in $[t_0-\tau, t_0]$. □

We observe that another proof of this result can be given, for instance, by means of Banach contractions theorem. However, the proof based on Grönwall lemma is more constructive. Indeed, Theorem 1.3 provides the unique solution of (1.1) in terms of the series

$$y(t) = y_0 + \sum_{n=0}^{\infty} (y_{n+1}(t) - y_n(t)),$$

where $y_n(t)$ is computed via the so-called *Picard iterations* (1.9). Clearly, this is only a formal representation of the solution, far from being assumable as a closed form for the solution of (1.1). Anyway, Picard iterations can be assumed as a primitive numerical methods to approximate the solution of (1.1) and the proof of

Theorem 1.3 provides some meaningful properties of this iterative method. First of all, Picard-Lindelöf proof shows that Picard iterations are convergent and an error estimate, computed as difference between two consecutive iterations, is given by Eq. (1.10). It is worth paying attention to the fact that (1.10) provides an upper bound for the error depending on the time window $t - t_0$, which requires to be compensated by a suitably large number n of iterations in order to gain an acceptable accuracy in the iterative process. Let us provide an example that clarifies this aspect.

Example 1.5 (Picard Iterations on a Linear Problem) We aim to provide an approximation to the solution of the linear problem

$$\begin{cases} y'(t) = \lambda y(t), & t \in [0, T], \\ y(0) = 1, \end{cases} \tag{1.11}$$

by means of Picard iterations. Specifically, we compute few Picard iterations and compare each of them with the exact solution of the problem, that is $y(t) = e^{\lambda t}$.

Let us compute the first four iterations:

$$\begin{aligned} y_1(t) &= 1 + \lambda \int_0^t y_0(s)ds = 1 + \lambda t, \\ y_2(t) &= 1 + \lambda \int_0^t y_1(s)ds = 1 + \lambda t + \frac{(\lambda t)^2}{2}, \\ y_3(t) &= 1 + \lambda \int_0^t y_2(s)ds = 1 + \lambda t + \frac{(\lambda t)^2}{2} + \frac{(\lambda t)^3}{6}, \\ y_4(t) &= 1 + \lambda \int_0^t y_3(s)ds = 1 + \lambda t + \frac{(\lambda t)^2}{2} + \frac{(\lambda t)^3}{6} + \frac{(\lambda t)^4}{24}. \end{aligned}$$

By induction, one can prove that the generic iteration $y_k(t)$ recovers the k-th partial sum of the power series expansion of $e^{\lambda t}$, i.e.

$$y_k(t) = 1 + \sum_{i=1}^{k} \frac{(\lambda t)^i}{i!}.$$

As proved in Theorem 1.3, the error estimate (1.10) depends on the time window. Clearly, larger is time window, bigger is the value of k we need to use for accuracy purposes, as shown in Table 1.1. As visible from the table, the error $|e^{\lambda T} - y_k(T)|$ is smaller if a proper balance between k and T is guaranteed, confirming the dependence on the time window proved in Theorem 1.3.

Table 1.1 Example 1.5: absolute errors $|e^{\lambda T} - y_k(T)|$ obtained applying k Picard iterations to Eq. (1.11), with various values of λ and T

λ	k	T	$\lvert e^{\lambda T} - y_k(T)\rvert$
-1	5	1	1.21e−03
	10	1	2.31e−08
	15	1	4.52e−14
	20	10	1.34e+01
	30	10	9.25e−04
	40	10	2.41e−09
1	5	1	1.61e−03
	10	1	2.73e−08
	15	1	5.06e−14
	20	10	3.50e+01
	30	10	1.76e−03
	40	10	3.92e−09

In summary, we have learned that existence and uniqueness of the solution to the initial value problem (1.1) rely on continuity and Lipschitz continuity of its vector field. We also need to stress that the hypothesis of Lipschitz continuity may be replaced by that of boundedness of $\partial f/\partial y$, clearly when this derivative exists. Indeed, if $\partial f/\partial y$ is uniformly bounded, then it is Lipschitz continuous; the proof of this property is left to the reader.

In order to recover Hadamard well-posedness, we finally need to provide a result of continuous dependence of the solution to (1.1) on the initial value and the vector field. Before presenting a rigorous proof, let us further elaborate on the meaning of this issue, which is very relevant also for the numerical approximation. We consider the following initial value problem

$$\begin{cases} \widetilde{y}'(t) = \widetilde{f}(t, \widetilde{y}(t)), \\ \widetilde{y}(t_0) = \widetilde{y}_0, \end{cases} \tag{1.12}$$

arising from the perturbation of the initial value and the vector field of (1.1), as follows:

$$\widetilde{y}_0 = y_0 + \delta_0, \qquad \widetilde{f}(t, \widetilde{y}(t)) = f(t, \widetilde{y}(t)) + \delta(t, \widetilde{y}(t)),$$

with $\delta_0 \in \mathbb{R}^d$ and $\delta : [t_0, T] \times \mathbb{R}^d \to \mathbb{R}^d$. We ask how far is the solution $y(t)$ of (1.1) from the solution $\widetilde{y}(t)$ of (1.12). Actually, when the third condition for Hadamard well posedness is fulfilled, perturbing the initial value and the vector field of (1.1) provides a perturbation to its solution of similar amplitude. Clearly, this concept has a strong connection with that of well-conditioning of the problem. Such an issue is of extreme importance in view of the numerical approximation of (1.1): indeed, anytime we look for computer solutions to a given problem, we always deal with its perturbation, since machine representation of real numbers is subject to round-off error.

Let us now analyze under which conditions the continuous dependence on the initial value and the vector field is guaranteed. To this purpose, we need to introduce a generalization of the aforementioned Grönwall lemmas, covering the case where the first summand in the right-hand side of (1.6) is no longer constant.

Lemma 1.3 (Generalized Grönwall Lemma) *Let $\alpha(t), \beta(t) : [t_0, \infty) \to \mathbb{R}$ be continuous functions, with $\beta(t) \geq 0$, for any $t \geq t_0$. If $y(t) : [t_0, \infty) \to \mathbb{R}$ is a continuous function such that*

$$y(t) \leq \alpha(t) + \int_{t_0}^{t} \beta(s) y(s) \mathrm{d}s, \quad t \geq t_0, \tag{1.13}$$

then,

$$y(t) \leq \alpha(t) + \int_{t_0}^{t} \alpha(s)\beta(s) \exp\left(\int_{s}^{t} \beta(\tau)\mathrm{d}\tau\right) \mathrm{d}s, \quad t \geq t_0.$$

Proof Let us define the auxiliary function

$$z(t) = \int_{t_0}^{t} \beta(s) y(s) \mathrm{d}s.$$

According to the fundamental theorem of calculus,

$$z'(t) = \beta(t) y(t)$$

and, by (1.13),

$$z'(t) - \beta(t) z(t) \leq \alpha(t)\beta(t).$$

We multiply each side of the last inequality by $\exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right)$, obtaining

$$\left(z'(t) - \beta(t) z(t)\right) \exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right) \leq \alpha(t)\beta(t) \exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right).$$

Taking into account that the left-hand side of this inequality is the first derivative of the function $w(t) = z(t) \exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right)$, we have

$$w'(t) \leq \alpha(t)\beta(t) \exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right), \quad t \geq t_0,$$

whose right-hand side is the first derivative of the function

$$v(t) = \int_{t_0}^{t} \alpha(s)\beta(s) \exp\left(-\int_{t_0}^{s} \beta(u)\mathrm{d}u\right) \mathrm{d}s.$$

As a consequence,

$$w'(t) - v'(t) \leq 0, \quad t \geq t_0$$

and the function $w(t) - v(t)$ is non-increasing for $t \geq t_0$, leading to

$$w(t) - v(t) \leq w(t_0) - v(t_0) = 0, \quad t \geq t_0.$$

Finally,

$$z(t) \exp\left(-\int_{t_0}^{t} \beta(s)\mathrm{d}s\right) \leq \int_{t_0}^{t} \alpha(s)\beta(s) \exp\left(-\int_{t_0}^{s} \beta(u)\mathrm{d}u\right) \mathrm{d}s, \quad t \geq t_0,$$

i.e.

$$z(t) \leq \int_{t_0}^{t} \alpha(s)\beta(s) \exp\left(\int_{s}^{t} \beta(u)\mathrm{d}u\right) \mathrm{d}s, \quad t \geq t_0,$$

that, matched with (1.13), gives the thesis. □

We can now state and prove the following result on the continuous dependence of the solution to (1.1) on the initial value and the vector field of the problem.

Theorem 1.4 *Let $y(t)$ and $z(t)$ respectively be solutions of the initial value problems*

$$\begin{cases} y'(t) = f(t, y(t)), \\ y(t_0) = y_0, \end{cases} \qquad \begin{cases} z'(t) = g(t, z(t)), \\ z(t_0) = z_0, \end{cases} \tag{1.14}$$

where $f, g : [t_0, T] \times D \to \mathbb{R}^d$, $D \subseteq \mathbb{R}^d$, are continuous functions. Suppose that f satisfies Lipschitz condition (1.8). Moreover, assume the existence of $\varepsilon > 0$ such that

$$\|f(t, w) - g(t, w)\| \leq \varepsilon, \qquad (t, w) \in [t_0, T] \times D.$$

Then,

$$\|y(t) - z(t)\| \leq \|y_0 - z_0\| \mathrm{e}^{L(t-t_0)} + \frac{\varepsilon}{L}\left(\mathrm{e}^{L(t-t_0)} - 1\right). \tag{1.15}$$

Proof Problems (1.14) are equivalent to

$$y(t) = y_0 + \int_{t_0}^{t} f(s, y(s))\mathrm{d}s, \quad t \in [t_0, T],$$
$$z(t) = z_0 + \int_{t_0}^{t} g(s, z(s))\mathrm{d}s, \quad t \in [t_0, T].$$

Side-by-side subtracting and passing to the norm yields

$$\|y(t) - z(t)\| \leq \|y_0 - z_0\| + \int_{t_0}^{t} \|f(s, y(s)) - g(s, z(s))\|\,\mathrm{d}s.$$

We add and subtract $f(s, z(s))$ in the norm of the last integrand, obtaining

$$\begin{aligned}\|y(t) - z(t)\| &\leq \|y_0 - z_0\| + \int_{t_0}^{t} \|f(s, y(s)) - f(s, z(s))\|\mathrm{d}s \\ &\quad + \int_{t_0}^{t} \|f(s, z(s)) - g(s, z(s))\|\mathrm{d}s \\ &\leq \|y_0 - z_0\| + L\int_{t_0}^{t} \|y(s) - z(s)\|\mathrm{d}s + \varepsilon(t - t_0).\end{aligned}$$

We are now in the typical situation where a Grönwall lemma is useful: indeed, we have an implicitly defined inequality on $\|y(t) - z(t)\|$. To make it explicit, we apply the generalized Grönwall lemma 1.3, with $\alpha(t) = \|y_0 - z_0\| + \varepsilon(t - t_0)$ and $\beta(t) = L$, obtaining

$$\begin{aligned}\|y(t) - z(t)\| &\leq \|y_0 - z_0\| + \varepsilon(t - t_0) \\ &\quad + L\int_{t_0}^{t} (\|y_0 - z_0\| + \varepsilon(s - t_0))\, e^{L(t-s)}\mathrm{d}s.\end{aligned} \tag{1.16}$$

We leave to the reader the computation of the integral appearing in the last inequality, which can be easily computed by parts, whose result is

$$\begin{aligned}&L\int_{t_0}^{t} \left(\|y_0 - z_0\| + \varepsilon(s - t_0)e^{L(t-s)}\mathrm{d}s\right) \\ &\qquad = \left(e^{L(t-t_0)} - 1\right)\left(\|y_0 - z_0\| + \frac{\varepsilon}{L}\right) - \varepsilon(t - t_0),\end{aligned}$$

that, included in (1.16), gives the thesis. □

We observe that inequality (1.15) allows us to conclude that, if the vector field of (1.1) is Lipschitz continuous, the initial values of (1.14) are close enough and ε is also small enough, the corresponding solutions $y(t)$ and $z(t)$ are also close enough to each other. In other terms, if we interpret z_0 as a perturbation of y_0 and g as a perturbation of f, problem (1.1) does not result to be so sensitive to such perturbations and they are not largely amplified on its solution. As aforementioned, this is a relevant property of the problem, especially in view of its numerical discretization.

According to the provided results, continuity of the vector field of a differential problem is an important ingredient for its Hadamard well-posedness. For the case of lack of continuity of the vector field, focusing on the autonomous case

$$\dot{y}(t) = f(y(t)), \qquad y(t_0) = y_0, \tag{1.17}$$

the interested reader can see, for instance, in [2, 8, 9, 75, 145–153, 165, 181–184, 230, 231] and references therein.

1.3 Dissipative Problems

The analysis of Hadamard well-posedness for (1.1) has revealed the importance of continuity and Lipschitz continuity of its vector field. Let us now discuss a variant of Lipschitz continuity, which looks particularly useful for the analysis of nonlinear problems (1.1). We start with a definition.

Definition 1.1 Consider any two solutions $y(t)$ and $\widetilde{y}(t)$ of the differential problem (1.1). The vector field $f(t, y(t))$ satisfies a *one-sided Lipschitz condition* if

$$\langle f(t, y(t)) - f(t, \widetilde{y}(t)), y(t) - \widetilde{y}(t) \rangle \leq \nu(t) \|y(t) - \widetilde{y}(t)\|^2, \quad t \in [t_0, T], \tag{1.18}$$

with respect to the scalar product $\langle \cdot\,, \cdot \rangle$, being $\| \cdot \|$ the corresponding induced norm.

The function $\nu(t)$ in (1.18) is usually denoted as *one-sided Lipschitz constant*, even if it is actually a function of the independent variable t.

Lipschitz continuous functions are certainly also one-sided Lipschitz, since by Schwarz inequality

$$\begin{aligned}\langle f(t,y)-f(t,\widetilde{y}(t)),y(t)-\widetilde{y}(t)\rangle &\le \|f(t,y)-f(t,\widetilde{y}(t))\|\,\|y(t)-\widetilde{y}(t)\|\\ &\le L\|y(t)-\widetilde{y}(t)\|^2,\end{aligned}$$

while the vice versa is not true in general. One-sided Lipschitz condition plays a role in the case of dissipative problems, i.e., problems generating contractive solutions, according to the following definition.

Definition 1.2 Consider any two solutions $y(t)$ and $\widetilde{y}(t)$ of (1.1), corresponding to the distinct initial values y_0 and $\widetilde{y}_0$, respectively. If, for a given norm $\|\cdot\|$, we have

$$\|y(t_2)-\widetilde{y}(t_2)\| \le \|y(t_1)-\widetilde{y}(t_1)\|,$$

for any t_1 and t_2 such that $t_0 \le t_1 \le t_2 \le T$, then we say that problem (1.1) is *dissipative* and generates *contractive solutions* in that norm.

The aforementioned link between the one-sided Lipschitz constant of the vector field and the generation of contractive solutions is clarified through the following result.

Theorem 1.5 *Consider any two solutions $y(t)$ and $\widetilde{y}(t)$ of the differential problem (1.1), corresponding to the distinct initial values y_0 and $\widetilde{y}_0$, respectively. Given a norm $\|\cdot\|$ induced by an inner product $\langle\cdot\,,\cdot\rangle$, if the vector field of (1.1) satisfies a one-sided Lipschitz condition (1.18) in that norm, with $\nu(t) \le 0$ for any $t \in [t_0, T]$, then the problem generates contractive solutions in the same norm.*

Proof Let us define the auxiliary function $g(t) = \|y(t)-\widetilde{y}(t)\|^2$ and compute its derivative. We obtain

$$\begin{aligned}g'(t) &= 2\langle y'(t)-\widetilde{y}'(t),y(t)-\widetilde{y}(t)\rangle\\ &= 2\langle f(t,y(t))-f(t,\widetilde{y}(t)),y(t)-\widetilde{y}(t)\rangle.\end{aligned}$$

Due to (1.18), we obtain the differential inequality

$$g'(t) \le 2\nu(t)g(t),$$

that can be handled in a similar way to the case of Grönwall lemmas presented in Sect. 1.2.

So, we multiply each side of the last equation by $\exp\left(-2\int_{t_0}^{t}\nu(s)\mathrm{d}s\right)$, getting

$$\left(g'(t)-2\nu(t)g(t)\right)\exp\left(-2\int_{t_0}^{t}\nu(s)\mathrm{d}s\right)\leq 0.$$

Taking into account that the left-hand side of this inequality is the first derivative of the function $g(t)\exp\left(-2\int_{t_0}^{t}\nu(s)\mathrm{d}s\right)$, we obtain that this function is non-increasing. Then, for any $t_0\leq t_1\leq t_2\leq T$,

$$g(t_2)\exp\left(-2\int_{t_0}^{t_2}\nu(s)\mathrm{d}s\right)\leq g(t_1)\exp\left(-2\int_{t_0}^{t_1}\nu(s)\mathrm{d}s\right)$$

or, equivalently, that

$$g(t_2)\leq g(t_1)\left[\exp\left(\int_{t_1}^{t_2}\nu(s)\mathrm{d}s\right)\right]^2,$$

i.e.,

$$\|y(t_2)-\widetilde{y}(t_2)\|\leq\|y(t_1)-\widetilde{y}(t_1)\|\exp\left(\int_{t_1}^{t_2}\nu(s)\mathrm{d}s\right)$$

and, since $\nu(t)\leq 0$, the thesis holds true. □

Computing the one-sided Lipschitz constant may be a highly non-trivial task and, as a consequence, it would be worth considering alternative conditions allowing a more effective application of Theorem 1.5. A way to handle this issue is related to the notion of *logarithmic norm*, introduced in 1958 through the independent contributions of Germund Dahlquist in his doctoral thesis [107] and by Lozinskii [255]. We also refer to [325] and for a presentation of the logarithmic norm framed within a historical overview.

Let us start presenting the definition of logarithmic norm for a constant matrix.

Definition 1.3 For a given matrix $A\in\mathbb{R}^{d\times d}$ and a given matrix norm $\|\cdot\|$, the *logarithmic norm* of A is

$$\mu(A)=\lim_{h\to 0^+}\frac{\|I+hA\|-1}{h},$$

being I the identity matrix in $\mathbb{R}^{d\times d}$ and $h>0$.

The logarithmic norm is not always positive, in fact it is not a norm (despite its name). This is the case, for instance, of negative definite matrices. Indeed, one can prove that, if the norm in Definition 1.3 is the 2-norm, then the corresponding logarithmic norm (denoted as $\mu_2(A)$) is the maximum eigenvalue of the matrix $(A + A^\intercal)/2$. As a consequence, if A is negative definite, the corresponding logarithmic norm is negative.

It is possible to prove that (1.1) generates contractive solutions if the logarithmic norm of the Jacobian of its vector field is non-positive [107]. The proof is here omitted, but it is useful to see this result applied to a simple example.

Example 1.6 Let us consider the following linear problem

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = A \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix}, \tag{1.19}$$

with

$$A = \begin{bmatrix} -\dfrac{5}{12} & \dfrac{125}{108} \\ -\dfrac{3}{5} & -\dfrac{5}{12} \end{bmatrix}.$$

The two distinct eigenvalues of the matrix $(A+A^\intercal)/2$ are given by $-751/1080$ and $-149/1080$. Hence, the logarithmic norm $\mu_2(A) \leq 0$ and (1.19) generates contractive solutions. In order to display this property, according to Definition 1.2, let us compare two solutions of (1.19), one of those being the trivial solution $\widetilde{y}(t) = 0$. We compute the analytical solution of (1.19) with initial value $y(0) = [1 \;\; 1]^\intercal$ (the complete calculation is left to the reader; see Exercise 5 in Sect. 1.6) and check if $\|y(t)\|_2$ is a non-increasing function. We depict $\|y(t)\|_2$ in Fig. 1.1, where its non-increasing monotonicity is visible, as expected.

1.4 Conservative Problems

In many practical situations, computing the solution of a given well-posed initial value problem (1.1) is not sufficient for an exhaustive understanding of the dynamics described by the problem itself. Indeed, solution related quantities may play a significant role in fully characterizing the problem and its role for the underlying applications. A relevant example is given by the energy conservation law: we know

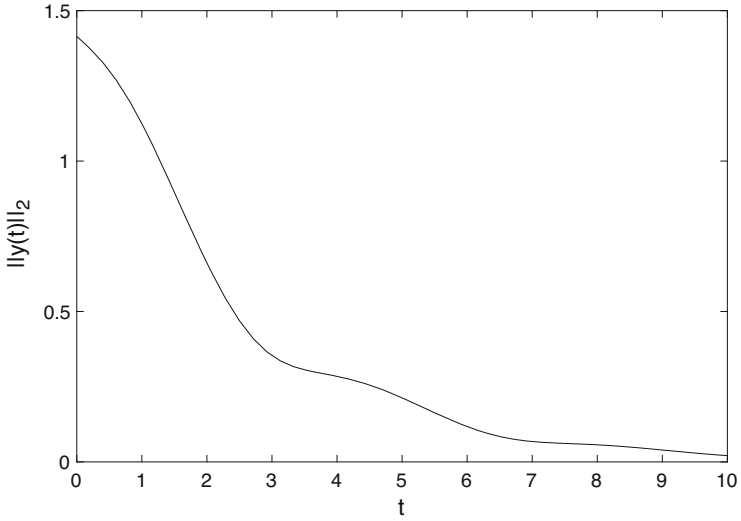


Fig. 1.1 Pattern of $\|y(t)\|_2$, where $y(t)$ is the solution of (1.19), with $y(0) = [1 \;\; 1]^{\mathsf{T}}$

that the total energy of an isolated system remains preserved over time. This is a characteristic property and, as we discuss in Chap. 8, it is important to make sure that such a preservation is guaranteed also along the numerical dynamics computed along approximate solutions of (1.1).

In the remainder of this section, we are mostly interested in quantities that remain preserved along the dynamics described by (1.1). Specifically, we focus our attention on a function $\mathcal{I}(y(t))$ (generally non-constant), such that

$$\mathcal{I}(y(t)) = \mathcal{I}(y(t_0)), \qquad t \in [t_0, T],$$

being $y(t)$ the solution to the autonomous problem (1.17). Such a function is usually called a *first integral* of (1.17). Clearly, since $\mathcal{I}(y(t))$ remains constant over solutions to (1.17), we have

$$0 = \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{I}(y(t)) = \nabla\mathcal{I}(y(t))y'(t) = \nabla\mathcal{I}(y(t))f(y(t)),$$

where $\nabla\mathcal{I}(y(t))$ is the gradient of $\mathcal{I}(y(t))$. The aforementioned arguments motivate the following definition.

Definition 1.4 For a given well-posed autonomous problem (1.17), a *first integral* is a function $\mathcal{I}(y(t))$ such that

$$\nabla \mathcal{I}(y(t))\, f(y(t)) = 0,$$

where $y(t)$ is the solution of (1.17).

Example 1.7 (Harmonic Oscillator) Let us consider a simple harmonic oscillator, i.e., a particle of unitary mass subject to a restoring force proportional to the displacement from its equilibrium position. The corresponding equation of motion is given by

$$y''(t) = -\omega^2 y(t),$$

where ω is the constant angular frequency of the oscillations. The equation is a second order linear ODE that can be regarded as the first order system

$$\begin{cases} y_1'(t) = y_2(t), \\ \\ y_2'(t) = -\omega^2 y_1(t). \end{cases} \tag{1.20}$$

It is well-known from classical mechanics that the total energy

$$E(y_1(t), y_2(t)) = \frac{1}{2} y_2^2(t) + \frac{1}{2}\omega^2 y_1^2(t) \tag{1.21}$$

is a first integral of (1.20). Let us prove it by applying Definition 1.4: by denoting $y(t) = [y_1(t) \;\; y_2(t)]^\intercal$, we have

$$\nabla \mathcal{I}(y(t))\, f(y(t)) = \begin{bmatrix} \omega^2 y_1(t) & y_2(t) \end{bmatrix} \begin{bmatrix} y_2(t) \\ -\omega^2 y_1(t) \end{bmatrix} = 0.$$

Figure 1.2 shows the pattern of the generated periodic orbit, for $\omega = 1$. Such a graph is given in the phase space, i.e., it is the set of couples $(y_1, y_2) \in \mathbb{R}^2$ satisfying (1.20).

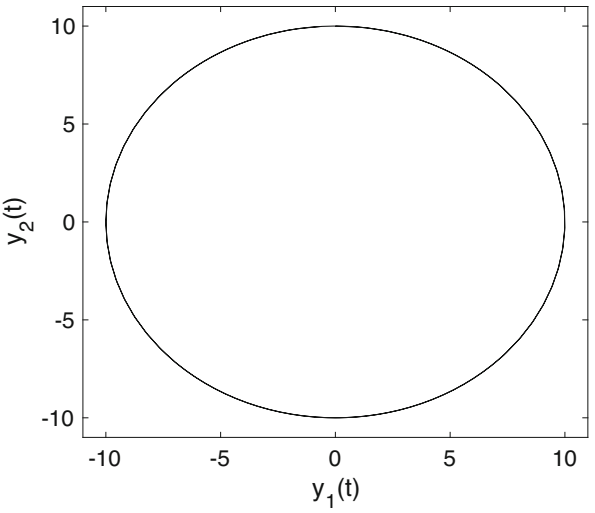


Fig. 1.2 Pattern of the orbit generated by the harmonic oscillator (1.20) in the phase space (y_1, y_2), with initial values $y_1(0) = 0$ and $y_2(0) = 10$, for $\omega = 1$

A relevant example of problem that maintains invariants along its solution is given by *Hamiltonian systems*, describing the dynamics of a system of d particles whose motion is characterized by their time varying generalized coordinates

$$q = [q_1(t) \;\; q_2(t) \;\; \ldots \;\; q_d(t)]^\intercal$$

and generalized momenta

$$p = [p_1(t) \;\; p_2(t) \;\; \ldots \;\; p_d(t)]^\intercal,$$

defined by

$$p_i = \frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}_i}, \quad i = 1, 2, \ldots, d,$$

where the function $\mathcal{L}(q, \dot{q}) = T(q, \dot{q}) - U(q)$ is the Lagrangian of the system (T is the kinetic energy, U is the potential energy). The corresponding equations of motion are given by

$$\begin{aligned} \dot{p}_i &= -\frac{\partial \mathcal{H}}{\partial q_i}(p, q), \\ & \qquad\qquad i = 1, 2, \ldots, d, \\ \dot{q}_i &= \frac{\partial \mathcal{H}}{\partial p_i}(p, q), \end{aligned} \tag{1.22}$$

denoted as *Hamilton equations*, where the dot stands for time derivative. The function $\mathcal{H}(p, q)$ is called *Hamiltonian function* and is linked to the Lagrangian

function according to the following relation

$$\mathcal{H}(p,q) = p^{\intercal}\dot{q} - \mathcal{L}(q,\dot{q}).$$

Let us prove that the Hamiltonian function is a first integral of (1.22). Indeed, according to Definition 1.4,

$$\nabla\mathcal{H}(p,q)\begin{bmatrix} -\dfrac{\partial\mathcal{H}}{\partial q_1}(p,q) \\ \vdots \\ -\dfrac{\partial\mathcal{H}}{\partial q_d}(p,q) \\ \dfrac{\partial\mathcal{H}}{\partial p_1}(p,q) \\ \vdots \\ \dfrac{\partial\mathcal{H}}{\partial p_d}(p,q) \end{bmatrix} = \sum_{i=1}^{d}\left(-\frac{\partial\mathcal{H}}{\partial p_i}(p,q)\frac{\partial\mathcal{H}}{\partial q_i}(p,q) + \frac{\partial\mathcal{H}}{\partial q_i}(p,q)\frac{\partial\mathcal{H}}{\partial p_i}(p,q)\right) = 0.$$

Example 1.8 (Mathematical Pendulum) We consider a specific example of Hamiltonian problem depending on a single degree of freedom ($d = 1$), modeling the motion of a particle of unitary mass constrained to a cord of negligible mass and unitary length, i.e., the so-called mathematical pendulum. The generalized coordinate q of the particle is the angle ϑ between the current position of the rod and the equilibrium position, while the generalized momentum p is the velocity $\dot{\vartheta}$. The corresponding Hamiltonian function assumes the form

$$\mathcal{H}(p,q) = \frac{p^2}{2} - \cos(q)$$

and, correspondingly, Eq. (1.22) takes the form

$$\begin{aligned} \dot{p} &= -\sin(q), \\ \dot{q} &= p. \end{aligned} \tag{1.23}$$

Figure 1.3 shows the pattern of the corresponding periodic orbit, displayed in the phase space (p, q).

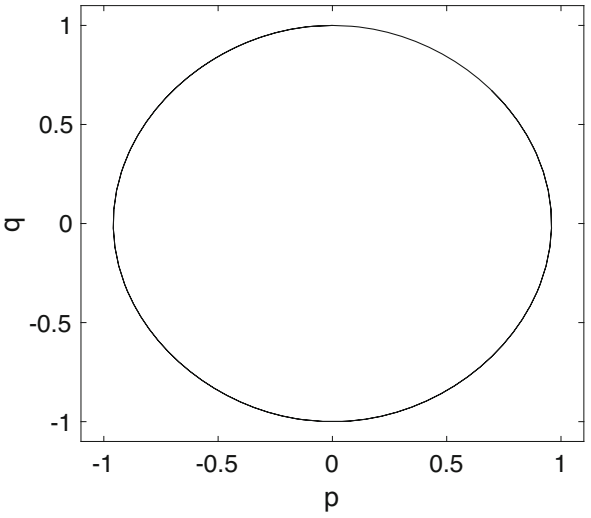


Fig. 1.3 Pattern of the orbit generated by (1.23), with initial values $y_1(0) = 0$ and $y_2(0) = 1$

We observe that the Hamiltonian function in Example 1.8 consists in the sum of two terms: one solely dependent on p, the other only on q. In this situation, the Hamiltonian function is given by the summation of kinetic and potential energies

$$\mathcal{H}(p, q) = T(p) + U(q), \tag{1.24}$$

separately depending on p and q. Hamiltonian functions as in Eq. (1.24) are denoted as *separable Hamiltonians* and the corresponding Hamiltonian problem (1.22) assumes the partitioned form

$$\dot{p}_i = -\frac{\mathrm{d}U}{\mathrm{d}q_i}, \qquad \dot{q}_i = \frac{\mathrm{d}T}{\mathrm{d}p_i}, \qquad i = 1, 2, \ldots, d. \tag{1.25}$$

As clarified in [192], separable Hamiltonian problems (1.25) allow a particularly efficient invariant preserving numerical approximation.

Example 1.9 (Hénon-Heiles Problem) An example of Hamiltonian problem depending on two degrees of freedom ($d = 2$), is the Hénon-Heiles model of the motion of stars around a galactic center, developed in 1964 by Michel Hénon and Carl Heiles [205] in Princeton University Observatory. The Hamiltonian function is given by

$$\mathcal{H}(p, q) = \frac{1}{2}\left(p_1^2 + p_2^2 + q_1^2 + q_2^2\right) + q_1^2 q_2 - \frac{1}{3} q_2^3 \tag{1.26}$$

(continued)

Example 1.9 (continued)
and, correspondingly, Eq. (1.22) assumes the form

$$\begin{aligned} \dot{p}_1 &= -q_1(1+2q_2), \\ \dot{p}_2 &= -q_1^2 + q_2^2 - q_2, \\ \dot{q}_1 &= p_1, \\ \dot{q}_2 &= p_2. \end{aligned} \tag{1.27}$$

Figure 1.4 displays the phase portrait in the plain (p_1, q_1). We finally observe that the Hamiltonian function (1.26) is separable and, as a consequence, the corresponding Hamiltonian problem (1.27) exhibits the usual partitioning as in (1.25).

We introduce the notation

$$y(t) = \begin{bmatrix} p(t) \\ q(t) \end{bmatrix}, \quad J = \begin{bmatrix} 0 & -I \\ I & 0 \end{bmatrix},$$

where $I \in \mathbb{R}^{d\times d}$ is the identity matrix and the other blocks of J are equal to the zero matrix of dimension $d \times d$. Correspondingly, Eq. (1.22) can be regarded in the compact form

$$\dot{y} = J\nabla\mathcal{H}(y). \tag{1.28}$$

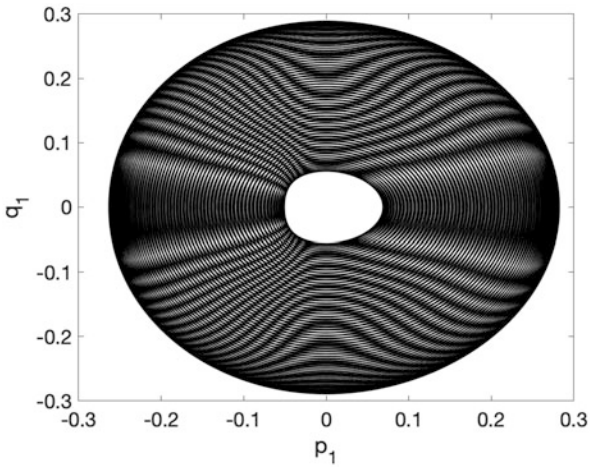


Fig. 1.4 Phase portrait of (1.27) in the plain (p_1, q_1), with initial values $p_1(0) = 0.2$, $p_2(0) = 0$, $q_1(0) = -0.2$, $q_2(0) = 0$

We now aim to prove a relevant property of Hamiltonian systems, which regards the structure of the associated flow. We remind that, for an autonomous problem (1.17) in $\mathbb{R}^d$ and for any $t \geq t_0$, the associated *flow map* $\Phi_t(y_0) : \mathbb{R}^d \to \mathbb{R}^d$ is given by

$$\Phi_t(y_0) = y(t).$$

In other terms, at any fixed $t \geq t_0$, the flow map associates the solution of (1.17) at time t. As a consequence,

$$\frac{\mathrm{d}}{\mathrm{d}t}\Phi_t(y_0) = f(\Phi_t(y_0)).$$

We now consider the Jacobian of each side and, as it regards the left-hand side, we swap time derivative and the derivative with respect to y, obtaining

$$\frac{\mathrm{d}}{\mathrm{d}t}\Phi_t'(y_0) = f'(\Phi_t(y_0))\Phi_t'(y_0)$$

and denoting $M = \Phi_t'(y_0)$ yields

$$\dot{M} = f'(\Phi_t(y_0))M. \tag{1.29}$$

This is a useful matrix differential equation describing the evolution of the Jacobian of the flow in time, known in the literature as *variational equation* associated to (1.17). We will use it in the proof of Poincaré theorem 1.6.

As aforementioned, we aim to provide a characteristic property of the flow map of Hamiltonian problems, for which we need to introduce the following definitions. For any two given vectors $\xi, \eta \in \mathbb{R}^{2d}$, we introduce the following bilinear form

$$\omega(\xi, \eta) = \sum_{i=1}^{d} \omega_i(\xi, \eta), \tag{1.30}$$

where

$$\omega_i(\xi, \eta) = \xi_i \eta_{d+i} - \xi_{d+i}\eta_i, \quad i = 1, 2, \ldots, d.$$

The form $\omega(\xi, \eta)$ has a geometric interpretation: since ξ and η span a bidimensional parallelogram in $\mathbb{R}^{2d}$, each $\omega_i(\xi, \eta)$ is the oriented area of the orthogonal projection of this parallelogram on the (q_i, p_i)-plane and $\omega(\xi, \eta)$ is the sum of such projected oriented areas. In a more compact notation, $\omega(\xi, \eta) = -\xi^{\intercal} J \eta$.

Definition 1.5 A given matrix $A \in \mathbb{R}^{2d\times 2d}$ is a *symplectic matrix* if it preserves the bilinear form (1.30), i.e.,

$$\omega(A\xi, A\eta) = \omega(\xi, \eta), \qquad \xi, \eta \in \mathbb{R}^{2d}.$$

In other terms, if A is a symplectic matrix,

$$\xi^{\intercal} A^{\intercal} J A \eta = \xi^{\intercal} J \eta$$

and, since this relation has to hold true for any $\xi, \eta \in \mathbb{R}^{2d}$, we have

$$A^{\intercal} J A = J, \tag{1.31}$$

that is the condition for the matrix A to be symplectic.

Example 1.10 Let us consider the following matrix

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}.$$

The reader can easily check that A is symplectic, since it satisfies condition (1.31). We now aim to check the preservation property given in Definition 1.5, i.e., $\omega(\xi, \eta) = \omega(A\xi, A\eta)$, for a given couple of vectors ξ and η. We consider the vectors ξ and η connecting the origin with the points of coordinate $(\frac{5}{2}, \frac{5}{2})$ and $(\frac{5}{2}, \frac{15}{2})$, respectively. The corresponding spanned parallelogram of area $\omega(\xi, \eta)$ is reported in Fig. 1.5, on the left. The transformed vectors $A\xi$ and $A\eta$, connecting the origin with the points $(5, \frac{5}{2})$ and $(10, \frac{15}{2})$ respectively, span the parallelogram of area $\omega(A\xi, A\eta)$ displayed in Fig. 1.5, on the right. Calculations left to the reader confirm that $\omega(\xi, \eta) = \omega(A\xi, A\eta) = \frac{25}{2}$.

The notion of symplecticity can be extended to any nonlinear map, according to the following definition.

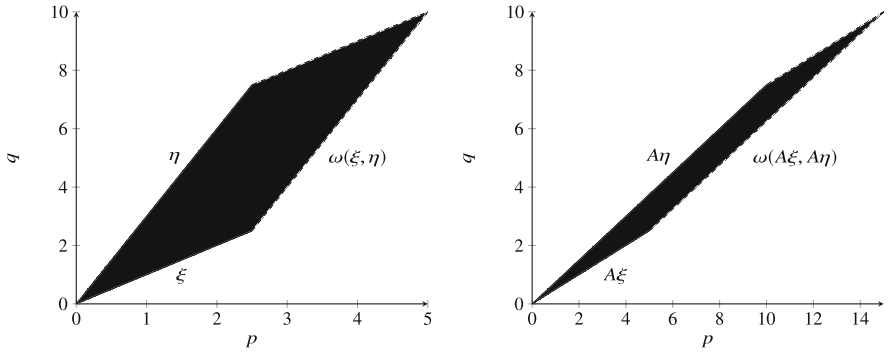


Fig. 1.5 Parallelograms spanned by the vectors ξ, η and the transformed ones $A\xi$, $A\eta$ through the symplectic matrix A, given in Example 1.10

Definition 1.6 A map $\alpha : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ is a *symplectic transformation* if, for any $y \in \mathbb{R}^{2d}$, the Jacobian $\alpha'(y)$ is a symplectic matrix, i.e.,

$$\alpha'(y)^{\mathsf{T}} J \alpha'(y) = J.$$

We are now able to prove the following fundamental result, highlighting a characteristic property of Hamiltonian problems.

Theorem 1.6 (Poincaré) *For any given Hamiltonian problem* (1.28)*, with* $\mathcal{H}(y)$ *twice continuously differentiable, the corresponding flow map* $\Phi_t(y_0)$ *is a symplectic transformation for any* t*, i.e.*

$$\Phi_t'(y_0)^{\mathsf{T}} J \Phi_t'(y_0) = J.$$

Proof We first specialize Eq. (1.29) to the case of Hamiltonian systems (1.28), whose vector field is given by

$$f(y) = J \nabla \mathcal{H}(y).$$

Consequently, its Jacobian assumes the form

$$f'(y) = J H_{yy},$$

where H_{yy} is the symmetric Hessian matrix of the second derivatives. Then, we obtain from (1.29) that

$$\dot{M} = J H_{yy} M.$$

Definition 1.6 requires the computation of $R = M^\intercal J M$. Let us differentiate R in time, obtaining

$$\dot{R} = \dot{M}^\intercal J M + M^\intercal J \dot{M} = M^\intercal H_{yy}^\intercal J^\intercal J M + M^\intercal J^2 H_{yy} M.$$

Since H_{yy} is symmetric, $J^2 = -I$ and $J^\intercal J = I$, we have

$$\dot{R} = M^\intercal H_{yy} M - M^\intercal H_{yy} M = 0.$$

Thus, R is constant in time. Since $R(0) = M(0)^\intercal J M(0) = J$, we have that R is constantly equal to its initial value J, that gives the thesis. □

1.5 Stability of Solutions

We now aim to focus on the stability properties of solutions of a d-dimensional differential problem (1.1), for $t \geq t_0$. In other terms, we briefly analyze the effects of perturbations to the initial value $y(t_0)$ on the solution of the problem. Moreover, we are also interested in analyzing asymptotic properties of solutions to (1.1), with a specific interest to the linear case. The interested reader can see, for instance, [14, 86, 99, 100, 199–201, 228, 245, 286] and references therein for a wider dedicated presentation of stability analysis.

Let us start providing the following useful definitions.

Definition 1.7 The solution $y(t)$ of a well-posed problem (1.1) is *stable* over the integration interval $[t_0, +\infty)$ if, for any $\varepsilon > 0$, there exists $\delta_\varepsilon > 0$ such that another solution $\widetilde{y}(t)$ of (1.1), obtained in correspondence of the initial value $\widetilde{y}_0$ satisfying

$$\|\widetilde{y}_0 - y_0\| < \delta_\varepsilon,$$

fulfills the inequality

$$\|\widetilde{y}(t) - y(t)\| < \varepsilon,$$

for any $t \geq t_0$, where $\|\cdot\|$ is a given norm.

In other terms, stable solutions are not sensitive to the effects of small perturbations to the initial value. According to Definition 1.7, this property is visible on the whole the integration interval. An analogous long-term property is now defined as follows.

Definition 1.8 The solution $y(t)$ of a well-posed problem (1.1) is *asymptotically stable* if it is stable and

$$\lim_{t\to\infty} \|\widetilde{y}(t) - y(t)\| = 0. \tag{1.32}$$

Hence, for asymptotically stable solutions of ODEs, the effects of perturbations to the initial value becomes more negligible the more t is bigger.

Considering an ODE system $y' = f(t, y)$, it is a custom to study the stability of the zero solution of an equivalent problem, obtained through the change of variables

$$x = y - \overline{y}(t), \tag{1.33}$$

supposing that $\overline{y}(t)$ is a given solution of the ODE. Side-by-side differentiation in (1.33) leads to

$$x' = g(t, x), \tag{1.34}$$

where $g(t, x) = f(t, x + \overline{y}(t)) - f(t, \overline{y}(t))$.

Let us now apply Definition 1.7: the zero solution of (1.34) is stable over the integration interval $[t_0, +\infty)$ if, for any $\varepsilon > 0$, there exists $\delta_\varepsilon > 0$ such that any solution $\widetilde{x}(t)$ of (1.34) satisfying $\|\widetilde{x}(t_0)\| < \delta_\varepsilon$ fulfills the inequality $\|\widetilde{x}(t)\| < \varepsilon$, for any $t \geq t_0$, where $\|\cdot\|$ is a given norm. Moreover, the zero solution of (1.34) is asymptotically stable if it is stable and $\|\widetilde{x}(t)\|$ tends to 0 as t tends to infinity. We observe that, when the zero solution of (1.34) is stable, we say that the system itself is stable; similarly, when the zero solution of (1.34) is asymptotically stable, we say that the system itself is asymptotically stable.

We now aim to provide a complete characterization of the stability of solutions to the linear problem

$$y'(t) = A(t)y(t) + b(t), \tag{1.35}$$

where the matrix $A(t) \in \mathbb{R}^{d\times d}$ and the vector $b(t) \in \mathbb{R}^d$ have continuous entries. By choosing a solution $y(t)$ of (1.35) and performing the change of variables (1.33), we have

$$x'(t) + \overline{y}'(t) = A(t)(x(t) + \overline{y}(t)) + b(t),$$

i.e.,

$$x'(t) = A(t)x(t), \tag{1.36}$$

whose zero solution $x(t) = 0$ corresponds to the chosen solution $y(t)$ of (1.35). The following result holds true.

Theorem 1.7 *The system (1.36) is stable if and only if there exists a constant $K > 0$, such that*

$$\|X(t)\| \leq K, \quad t \geq t_0 \tag{1.37}$$

and it is asymptotically stable if and only if

$$\lim_{t\to\infty} \|X(t)\| = 0, \tag{1.38}$$

being $X(t)$ the fundamental matrix of (1.36), i.e., the matrix with linearly independent columns such that $X'(t) = A(t)X(t)$.

Proof We assume, without loss of generality, that $X(t_0) = I$, where I is the identity matrix in $\mathbb{R}^{d\times d}$. Then, by denoting $x(t_0) = x_0 \neq 0$, the solution $x(t)$ of (1.36) is given by

$$x(t) = X(t)x_0,$$

as it can be checked by the reader through its direct replacement in (1.36).

- We first provide the stability proof. Let us assume that (1.37) holds true. Then,

  $$\|x(t)\| \leq \|X(t)\|\|x_0\| \leq K\|x_0\|.$$

  Hence, the stability inequality $\|x(t)\| < \varepsilon$ holds true supposing that $\|x_0\| < \delta_\varepsilon = \frac{\varepsilon}{K}$. This proves that the system is stable. Let us now prove that stability implies (1.37). The stability hypothesis suggests that, for any $\varepsilon > 0$, there exists $\delta_\varepsilon > 0$ with $\|x_0\| < \delta_\varepsilon$, such that $\|X(t)x_0\| < \varepsilon$. Then,

  $$\|X(t)\| = \sup_{x\neq 0} \frac{\|X(t)x\|}{\|x\|} = \frac{1}{\delta_\varepsilon} \sup_{x\neq 0} \left\| X(t) \frac{\delta_\varepsilon x}{\|x\|} \right\| = \frac{1}{\delta_\varepsilon} \sup_{\|\eta\|=\delta_\varepsilon} \|X(t)\eta\|$$

  and the thesis $\|X(t)\| \leq K$ holds true for any $K < \frac{\varepsilon}{\delta_\varepsilon}$.
- We finally provide the asymptotic stability proof. Let us assume (1.38) satisfied. Since $\|x(t)\| \leq \|X(t)\|\|x_0\|$, (1.32) immediately holds true, as t tends to infinity.

Let us now prove that asymptotic stability implies (1.38). We have already proved the following expression for $\|X(t)\|$:

$$\|X(t)\| = \frac{1}{\delta_\varepsilon} \sup_{\|\eta\|=\delta_\varepsilon} \|X(t)\eta\| .$$

The asymptotic stability hypothesis suggests that, for any $\varepsilon > 0$, there exists $\delta_\varepsilon > 0$ with $\|x_0\| < \delta_\varepsilon$, such that $\|x(t)\|$ goes to 0 as t tends to infinity. As a consequence, there exists $\overline{\eta}$, with $\|\overline{\eta}\| = \delta_\varepsilon$, such that

$$\|X(t)\| = \frac{1}{\delta_\varepsilon} \|X(t)\overline{\eta}\| .$$

Then, $\|X(t)\|$ goes to 0 when t tends to infinity, since it inherits the same behavior from above right-hand side. □

It is also interesting to analyze what happens when the matrix $A(t)$ in (1.36) is identically equal to a constant matrix $A \in \mathbb{R}^{d\times d}$, i.e., when the system assumes the form

$$x'(t) = Ax(t). \tag{1.39}$$

This case is covered by the following result.

Theorem 1.8 *The system (1.39) is stable if and only if any eigenvalue of the matrix A has non-positive real part and those with zero real parts are simple. The system is asymptotically stable if and only if any eigenvalue of the matrix A has negative real part.*

The proof is here omitted, but the interested reader can find it, for instance, in [99, 228]. We observe that the conditions on the spectrum of A are not applicable when the matrix is time-dependent. A relevant counterexample has been given by Dekker and Verwer [141]. Consider the matrix

$$A(t) = \begin{bmatrix} -1 - 9\cos^2(6t) + 6\sin(12t) & 12\cos^2(6t) + \frac{9}{2}\sin(12t) \\ -12\sin^2(6t) + \frac{9}{2}\sin(12t) & -1 - 9\sin^2(6t) - 6\sin(12t) \end{bmatrix},$$

whose eigenvalues are -1 and -10 for any t, but the fundamental matrix of the corresponding system (1.36) is

$$X(t) = \begin{bmatrix} e^{2t}(\cos(6t) + 2\sin(6t)) & e^{-13t}(\sin(6t) - 2\cos(6t)) \\ e^{2t}(2\cos(6t) - \sin(6t)) & e^{-13t}(2\sin(6t) + \cos(6t)) \end{bmatrix}$$

and, by Theorem 1.7, the system is not stable.

Example 1.11 We aim to analyze stability and asymptotic stability properties of the system

$$\begin{bmatrix} x_1'(t) \\ x_2'(t) \end{bmatrix} = \frac{1}{t^2} \begin{bmatrix} 0 & t^2 \\ 2 & -3t \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, \quad t \geq 1. \tag{1.40}$$

In order to provide a fundamental matrix for the system, we look for a couple of linearly independent solutions. We observe that (1.40) is equivalent to the second order differential equation

$$x_1''(t) + \frac{3}{t} x_1'(t) - \frac{2}{t^2} x_1(t) = 0.$$

Let us look for solutions of the form $x_1(t) = t^n$. Then

$$n(n-1)t^{n-2} + \frac{3}{t} n t^{n-1} - \frac{2}{t^2} t^n = 0,$$

leading to $n^2 + 2n - 2 = 0$. This quadratic equation has two distinct solutions, i.e., $n = -1$ and $n = 3$, providing the following linearly independent solutions of (1.40):

$$x(t) = \frac{1}{t^2} \begin{bmatrix} t^2 \\ -1 \end{bmatrix}, \quad x(t) = t^2 \begin{bmatrix} t \\ 3 \end{bmatrix}.$$

As a consequence, a fundamental matrix for (1.40) is given by

$$X(t) = \frac{1}{t^2} \begin{bmatrix} t & t^5 \\ -1 & 3t^4 \end{bmatrix}$$

and, according to Theorem 1.7, system (1.40) is neither stable nor asymptotically stable.

1.6 Exercises

1. Study existence and uniqueness of the solution to the following initial value problem

$$\begin{cases} y'(t) = y^2(t), & t \in [0, 2], \\ y(0) = 0. \end{cases}$$

   Moreover, provide a closed form of Picard iterations (1.9) associated to this problem.
2. Study existence and uniqueness of the solution to the following initial value problem

$$y'(t) = \begin{cases} -\dfrac{3}{4}, & t \in [0, 5), \\ \dfrac{1}{4}, & t = 5, \\ \dfrac{3}{4}, & t \in (5, 10], \end{cases}$$
$$y(0) = 0.$$

   What happens if $y'(t) = 0$, when $t = 5$?
3. Compute the first five Picard iterations associated to the following scalar initial value problem

$$\begin{cases} y'(t) = -2ty(t), & t \in [0, 10], \\ y(0) = 1. \end{cases}$$

4. Write a software in your chosen programming language that computes the solution of a scalar initial value problem with a certain prescribed accuracy, by means of Picard iterations (1.9). Each iteration requires the approximation of the integral in (1.9) via a certain chosen quadrature formula. For instance, you might use the trapezoidal rule

$$\int_a^b f(x)dx \approx (b-a)\frac{f(a)+f(b)}{2}, \tag{1.41}$$

   or its composite version

$$\int_a^b f(x)dx \approx \frac{b-a}{M}\sum_{k=1}^{M}\left(f(x_{k+1})+f(x_k)\right), \tag{1.42}$$

where M is the number of subintervals of equal length in which you divide the interval $[a, b]$. Which stopping criterion for the iterative process would you implement? Do you observe any difference if the composite quadrature rule (1.42) is used instead of the simple one (1.41)?

5. Compute the analytical solution of (1.19) and prove that its 2-norm is non-increasing.
6. Consider the following Lennard-Jones oscillator

$$\begin{cases} y_1'(t) = y_2(t), \\ \\ y_2'(t) = 12(y_1^{-13}(t) - y_1^{-7}(t)), \end{cases}$$

for $t \geq 0$ and prove that its total energy is a first integral of the system.

7. Given the linear system $x'(t) = Ax(t)$, $t \geq 0$, with

$$A = \begin{bmatrix} 1 & -1 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 & 2 \\ \frac{2}{3} & \frac{1}{2} & -1 & 0 \\ 2 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix},$$

analyze its stability and asymptotic stability properties.

8. Analyze stability and asymptotic stability properties of $x'(t) = A(t)x(t)$, $t \geq 1$, where

$$A(t) = \frac{1}{t^2} \begin{bmatrix} 0 & t^2 \\ 1 & -t \end{bmatrix}.$$

9. Considering the SIR model for fake news diffusion (1.3), compute its linearization around a chosen vector of initial values and analyze the spectrum of the Jacobian of the linearized vector field. Then, compute the corresponding set of moduli of the eigenvalues and the ratio between the maximum and the minimum of this set, in correspondence of the values of the parameters α and β reported in Table 1.2 (also refer to [137]). Finally, comment on the dynamics of fake news in the countries listed in Table 1.2, by using the information arising from the computed values of the ratios.

Table 1.2 Values of the constants α, β in (1.3), for France, India, Italy, Mexico and United States, referring to 2019

Country	α	β
France	0.009	0.089
India	0.006	0.035
Italy	0.009	0.061
Mexico	0.008	0.064
United States	0.009	0.075

Chapter 2
Discretization of the Problem

This book is devoted to a subject – the numerical solution of ordinary differential equations – where practical relevance meets mathematical beauty in a unique way.

(Jesús María Sanz-Serna, Foreword to the book of John C. Butcher [67])

The introduction to differential problems, delivered in Chap. 1, has established a number of useful issues to start our trip to the core topic of the book: how to compute approximate solutions to (1.1). In this chapter we aim to introduce some basic concepts characterizing the approximation of (1.1), as well as basic requirements that a numerical method has to fulfill. Clearly, all results only apply to Hadamard well-posed initial value problems (1.1), in view of the approximation to their unique solution.

2.1 Domain Discretization

The solution of (1.1) is a function, i.e., a continuous object, therefore one cannot expect to provide its analytical expression as output of computer calculations (unless it is provided by symbolic computations, that do not fall within the scopes of this book). As a consequence, since we aim to provide approximate solutions in a computing environment working with a finite arithmetic, it is necessary to understand how to define a numerical solution of (1.1). The numerical approximation of (1.1) requires a preliminary step of transforming the continuous problem (1.1) into its *discretized* counterpart. Then, numerically solving (1.1) is the process of providing an accurate approximation of its solution computed in a discrete set of sampled points. There are certainly many ways to numerically solve (1.1), both of general type or specifically tailored to problem in order to reproduce significant properties of the continuous problem along its discretized dynamics.

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_2

The process of discretization of (1.1) requires two fundamental steps:

- discretizing the domain of (1.1), i.e., the interval $[t_0, T]$;
- discretizing the differential equation itself.

In this section we discuss the first issue. Discretizing the domain of the problem, i.e. the interval $I = [t_0, T]$, requires detecting a number of points belonging to I; correspondingly, a numerical method provides an approximate value of the solution of (1.1) in these selected points. The most common way to discretize the interval $[t_0, T]$ consists in dividing it in a number of subintervals of equal lengths. In other terms, let us introduce the following set of $N + 1$ equidistant points

$$\mathcal{I}_h = \{t_0 < t_1 < t_2 < \cdots < t_N = T, \quad N = (T - t_0)/h\}, \tag{2.1}$$

so that the interval I is divided in N subintervals of equal length h. Then, the generic point $t_n \in \mathcal{I}_h$ assumes the form

$$t_n = t_0 + nh, \quad n = 0, 1, \ldots, N.$$

Each element of $\mathcal{I}_h$ is usually denoted as *grid point* and h is the *fixed stepsize* of the grid $\mathcal{I}_h$. A numerical method for (1.1) is designed in order to compute the set

$$\{y_n \approx y(t_n),\ t_n \in \mathcal{I}_h,\ n = 0, 1, \ldots, N\} \tag{2.2}$$

of approximate values of the solution $y(t)$ in the grid points. The following sections and chapters are specifically dedicated to presenting how the computation of such a set of values is performed.

Grid points may even be non-equidistant and their distribution can be adapted to the behavior of the solution: in this case, we talk about *variable stepsize* grids

$$\mathcal{I}_{|h|} = \{t_0 < t_1 < t_2 < \cdots < t_N = T,\ t_{n+1} = t_n + h_n,\ n = 0, 1, \ldots, N - 1\},$$

where

$$|h| = \max_{n=0,1,\ldots,N-1} h_n$$

is the *fineness* of the grid. In the remainder of the presentation, unless differently specified, we normally consider fixed stepsize grids. Variable stepsize environments and technique of adaptive stepsize selection are described in Chap. 7.

2.2 Difference Equations: The Discrete Counterpart of Differential Equations

The original continuous problem (1.1) is based on a differential system whose solution is defined for any $t \in [t_0, T]$. On the other side, the approximate solution (2.2) is defined in correspondence of the set of grid points $\mathcal{I}_h$ defined by (2.1). As a consequence, the discrete counterpart of a differential equation is given by a functional relation involving the values of the numerical solution (2.2). In other terms, the discretized version of (1.1) is a *difference equation*.

Let us briefly introduce some basic ideas on difference equations; for a more detailed presentation of the topic, the interested reader can refer, for instance, to [3, 160, 177, 235, 241, 297] and references therein.

Definition 2.1 For a non-negative integer number n_0, a *difference equation of order k* in the unknowns $\{y_n\}_{n \geq n_0}$, with $y_n \in \mathbb{R}^d$ for any $n \geq n_0$, is given by the functional relation

$$F(y_n, y_{n+1}, \ldots, y_{n+k}) = 0, \quad n \geq n_0. \tag{2.3}$$

Let us promptly dispel a possible misunderstanding: do not think that (2.3) is simpler to be solved than (1.1). Indeed, exact solutions of (2.3) can be computed in closed form only in very specific, simple cases, e.g., for linear equations. General procedures to solve any nonlinear difference equation are not available in the existing literature. However, passing from differential to difference equations allows us to activate, as we will see, an effective *step by step* procedure to compute approximate solutions. This aspect will be made more explicit later.

2.2.1 Linear Difference Equations

As announced, we are able to compute, through a general procedure, only the solutions of linear difference equations. Hence, it is now worth putting some efforts into presenting such a solving procedure; our attention is here specifically addressed to the case of scalar linear equations with constant coefficients and order k, i.e.,

$$\alpha_k y_{n+k} + \alpha_{k-1} y_{n+k-1} + \cdots + \alpha_0 y_n = \beta_n, \quad n \geq n_0, \tag{2.4}$$

with $\alpha_i \in \mathbb{R}$, $i = 0, 1, \ldots, k$, and $\beta_n \in \mathbb{R}$. The coefficient α_k can be assumed equal to 1, without loss of generality (indeed, it is always possible to normalize all coefficients in order to fall in this case).

The case of systems of linear difference equations is here omitted, but the reader can find their detailed presentation in specific monographs on the theory of difference equations, such as the beautiful book [241] by V. Lakshmikantham (India, 1924-Melbourne, 2012) and D. Trigiante (Laterza, 1944-Lido di Camaiore, 2011).

We first aim to discuss about the existence and uniqueness of solutions of (2.4), equipped by a proper number of initial conditions. To this purpose, let us suppose that (2.4) is equipped by the following k initial conditions

$$y_{n_0} = c_0, \quad y_{n_0+1} = c_1, \quad \ldots, \quad y_{n_0+k-1} = c_{k-1}, \tag{2.5}$$

with $c_i \in \mathbb{R}$, $i = 0, 1, \ldots, k-1$. According to Definition 2.1, a solution of Eq. (2.4) is a sequence $\{y_n\}_{n \geq n_0}$ satisfying (2.4) for any $n \geq n_0$. We aim to prove that, supplying (2.4) with the set of initial values (2.5), the following existence and uniqueness result holds true.

Theorem 2.1 *Problem (2.4), equipped by the set of initial conditions (2.5), has a unique solution $\{y_n\}_{n \geq n_0}$ such that*

$$y_{n_0} = c_0, \quad y_{n_0+1} = c_1, \quad \ldots, \quad y_{n_0+k-1} = c_{k-1}.$$

Proof Since $\alpha_k \neq 0$, replacing (2.5) in (2.4) leads to a linear algebraic equation in y_{n_0+k}, allowing to compute it in unique way. Replacing $y_{n_0+1}, y_{n_0+2}, \ldots, y_{n_0+k}$ in (2.4) permits to compute y_{n_0+k+1} in unique way as well. Thus proceeding, each value of y_n, for any $n \geq n_0$, can be uniquely computed. □

By defining the operator

$$\mathcal{L}(y_n) = \sum_{i=0}^{k} \alpha_i y_{n+i}, \tag{2.6}$$

we can recast (2.4) in the following operator form

$$\mathcal{L}(y_n) = \beta_n. \tag{2.7}$$

Clearly, $\mathcal{L}(\cdot)$ is a linear operator, since, for any $a, b \in \mathbb{R}$ and any sequence $\{y_n\}_{n \geq n_0}$, $\{z_n\}_{n \geq n_0}$,

$$\mathcal{L}(a y_n + b z_n) = \sum_{i=0}^{k} \alpha_i (a y_{n+i} + b z_{n+i}) = a\mathcal{L}(y_n) + b\mathcal{L}(z_n).$$

Let us now distinguish the case of homogeneous and inhomogeneous scalar linear difference equations, with the aim to provide a representation formula for the solution in both cases.

2.2.2 Homogeneous Case

As a consequence of the linearity of operator (2.6), if we denote with S the set of solutions of the homogeneous equation

$$\mathcal{L}(y_n) = 0, \tag{2.8}$$

we promptly discover that any linear combination of elements of S still lies in S.

We now aim to provide a representation formula for the element of S. To this purpose, we need to start with the following useful lemma.

Lemma 2.1 *Let us suppose that $\{y_n\}_{n\geq n_0}$ is solution of the order k homogeneous difference equation (2.8) with respect to a given vector of initial conditions $c = (c_\ell)_{\ell=0}^{k-1}$, i.e.,*

$$y_{n_0} = c_0, \quad y_{n_0+1} = c_1, \quad \dots, \quad y_{n_0+k-1} = c_{k-1}.$$

Let us also assume that, for any $i = 1, 2, \dots, k$, the sequence $\{y_n^i\}_{n\geq n_0}$ is solution of (2.8), with respect to the vector of initial conditions $e_i \in \mathbb{R}^k$ given by the i-th vector of the canonical basis of $\mathbb{R}^k$ (hence, $e_{ij} = \delta_{ij}$, where δ_{ij} is the Kronecker delta, for $i, j = 1, 2, \dots, k$). Then, for any $n \geq n_0$,

$$y_n = \sum_{i=1}^{k} c_{i-1} y_n^i.$$

Proof The proof is constructive. Let us introduce the auxiliary sequence $\{z_n\}_{n\geq n_0}$, with

$$z_n = \sum_{i=1}^{k} c_{i-1} y_n^i, \quad n \geq n_0.$$

The sequence $\{z_n^i\}_{n\geq n_0}$ is linear combination of elements of S, so it still lies in S. Its first k values are given by

$$\begin{aligned} z_{n_0} &= \sum_{i=1}^{k} c_{i-1} y_{n_0}^i = c_0, \\ z_{n_0+1} &= \sum_{i=1}^{k} c_{i-1} y_{n_0+1}^i = c_1, \\ &\vdots \\ z_{n_0+k-1} &= \sum_{i=1}^{k} c_{i-1} y_{n_0+k-1}^i = c_{k-1}. \end{aligned}$$

Then, $\{z_n\}_{n\geq n_0}$, is solution of (2.8) with respect to the vector of initial conditions c, exactly as $\{y_n\}_{n\geq n_0}$. The thesis holds true due to the uniqueness of the solution stated by Theorem 2.1, that gives $y_n = z_n$, for any $n \geq n_0$. □

In other words, Lemma 2.1 states that any solution of (2.8) can be represented as linear combination of the k solutions of (2.8)

$$\left\{\{y_n^1\}_{n\geq n_0},\ \{y_n^2\}_{n\geq n_0},\ \ldots,\ \{y_n^k\}_{n\geq n_0}\right\}, \tag{2.9}$$

obtained with respect to the k vectors of initial values given by the canonical basis of $\mathbb{R}^k$. We now aim to prove that the system of generators (2.9) of S is a set of linearly independent sequences, according to the following definition.

Definition 2.2 Let n_0 be a given non-negative integer number. k given sequences of scalars

$$\left\{\{f_n^1\}_{n\geq n_0},\ \{f_n^2\}_{n\geq n_0},\ \ldots,\ \{f_n^k\}_{n\geq n_0}\right\} \tag{2.10}$$

are *linearly independent* if, for any $n \geq n_0$, having

$$\sum_{i=1}^{k} \sigma_i f_n^i = 0$$

implies $\sigma_i = 0$, for any $i = 1, 2, \ldots, k$.

Definition 2.3 The *Casorati matrix* K_n associated to the set (2.10) is given by

$$K_n = \begin{bmatrix} f_n^1 & f_n^2 & \cdots & f_n^k \\ f_{n+1}^1 & f_{n+1}^2 & \cdots & f_{n+1}^k \\ \vdots & \vdots & \ddots & \vdots \\ f_{n+k-1}^1 & f_{n+k-1}^2 & \cdots & f_{n+k-1}^k \end{bmatrix} \in \mathbb{R}^{k\times k}. \tag{2.11}$$

The Casorati matrix is a useful tool for the analysis of linear independence, according to the following result.

Theorem 2.2 *Let n_0 be a given non-negative integer number. Consider k given sequences (2.10) and denote by K_n the corresponding Casorati matrix (2.11). If there exists $\overline{n} \geq n_0$ such that $\det K_{\overline{n}} \neq 0$, then the sequences (2.10) are linearly independent.*

Proof Let us consider the following homogeneous linear system of k algebraic equations

$$\sum_{i=1}^{k} \sigma_i f_{n+\ell}^i = 0, \quad \ell = 0, 1, \ldots, k-1,$$

in the unknowns $\sigma_1, \sigma_2, \ldots, \sigma_k$, whose coefficient matrix is the Casorati matrix K_n given by (2.11). Since, for $n = \overline{n}$, $\det K_{\overline{n}} \neq 0$ by hypothesis, then the unique solution of above system is $\sigma_1 = \sigma_2 = \cdots = \sigma_k = 0$, which means that the sequences (2.10) are linearly independent. □

If (2.10) is a set of solution of (2.8), we can assume $\overline{n} = n_0$ in Theorem 2.2. Indeed, in this case, $\det K_{n_0} \neq 0$ implies that $\det K_n \neq 0$, for any $n \geq n_0$. This holds true because

$$\det K_{n_0+1} = (-1)^k \alpha_0 \det K_{n_0}. \tag{2.12}$$

The proof is left to the reader, here we only give few examples for specific values of k. If $k = 1$, the corresponding homogeneous equation (2.8) is

$$f_{n_0+1}^1 + \alpha_0 f_{n_0}^1 = 0,$$

and, therefore,

$$K_{n_0+1} = f^1_{n_0+1} = -\alpha_0 f^1_{n_0} = -\alpha_0 K_{n_0},$$

leading to (2.12). If $k = 2$, the corresponding homogeneous equation (2.8) is

$$f^i_{n_0+2} + \alpha_1 f^i_{n_0+1} + \alpha_0 f^i_{n_0} = 0, \quad i = 1, 2.$$

Then, the Casorati matrices involved in the corresponding Eq. (2.12) are

$$K_{n_0+1} = \begin{bmatrix} f^1_{n_0+1} & f^2_{n_0+1} \\ f^1_{n_0+2} & f^2_{n_0+2} \end{bmatrix}, \quad K_{n_0} = \begin{bmatrix} f^1_{n_0} & f^2_{n_0} \\ f^1_{n_0+1} & f^2_{n_0+1} \end{bmatrix}.$$

Therefore, Eq. (2.12) is satisfied, since

$$\begin{aligned} \det K_{n_0+1} &= f^1_{n_0+1} f^2_{n_0+2} - f^2_{n_0+1} f^1_{n_0+2} \\ &= f^1_{n_0+1} \left(-\alpha_1 f^2_{n_0+1} - \alpha_0 f^2_{n_0}\right) - f^2_{n_0+1} \left(-\alpha_1 f^1_{n_0+1} - \alpha_0 f^1_{n_0}\right) \\ &= \alpha_0 \left(f^1_{n_0} f^2_{n_0+1} - f^2_{n_0} f^1_{n_0+1}\right) = \alpha_0 \det K_{n_0}. \end{aligned}$$

As a consequence of Theorem 2.2 with $\overline{n} = n_0$, we have that the sequences in (2.9) are linearly independent, since the corresponding Casorati matrix K_{n_0} is the identity matrix in $\mathbb{R}^{k\times k}$. Hence, the set (2.9) is a basis for the space S of all solutions of (2.8), that results to be a linear space of dimension k. The basis (2.9) is called *canonical basis* of S.

In conclusion, we can state that the linear combination of k linearly independent solutions of (2.8) provides its general solution.

2.2.3 *Inhomogeneous Case*

We now complete our analysis by describing how to obtain the general solution of the inhomogeneous case (2.7). The reader will find a certain analogy with the case of linear ODEs.

Theorem 2.3 *Given a system of k linearly independent solutions $\{f_n^i\}_{n\geq n_0}$, $i = 1, 2, \ldots, k$, of the homogeneous equation (2.8) and an arbitrary solution $\{\overline{y}_n\}_{n\geq n_0}$ of the corresponding inhomogeneous problem (2.7), then the general solution of (2.7) is given by*

$$y_n = \overline{y}_n + \sum_{i=1}^{k} \sigma_i f_n^i, \quad n \geq n_0.$$

Proof Since

$$y_n - \overline{y}_n = \sum_{i=1}^{k} \sigma_i f_n^i, \quad n \geq n_0,$$

we have that $y_n - \overline{y}_n$ is solution of (2.8), i.e., $\mathcal{L}(y_n - \overline{y}_n) = 0$. The linearity of $\mathcal{L}$ leads to

$$\mathcal{L}(y_n) = \mathcal{L}(\overline{y}_n) = \beta_n$$

and the thesis holds true. □

Hence, the general solution of (2.7) is obtained as sum of the general solution of (2.8) plus an arbitrary particular solution of (2.7). As a consequence, the first step to perform is finding k linearly independent solutions of (2.8). As in the case of the analytical solution of linear ODEs, we look for solutions of the form $y_n = x^n$, $n \in \mathbb{N}$, with $x \neq 0$. Replacing this ansatz in (2.8) yields

$$x^{n+k} + \alpha_{k-1}x^{n+k-1} + \ldots + \alpha_0 x^n = 0,$$

i.e.,

$$x^k + \alpha_{k-1}x^{k-1} + \ldots + \alpha_0 = 0.$$

In other terms, x is solution of the characteristic polynomial

$$\rho(x) = x^k + \alpha_{k-1}x^{k-1} + \ldots + \alpha_0 \tag{2.13}$$

of (2.8). Clearly, if ξ is a solution of (2.13), then $y_n = \xi^n$ is solution of (2.8). We distinguish three possible cases.

- Case 1: the characteristic polynomial (2.13) has k distinct roots ξ_1, ξ_2, …, ξ_k. As a consequence, we have k linearly independent solutions $\{\xi_i^n\}_{n\in\mathbb{N}}$,

$i = 1, 2, \ldots, k$. Their linear independence can be easily obtained by inspecting the Casorati matrix

$$K_0 = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \xi_1 & \xi_2 & \ldots & \xi_{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ \xi_1^{k-1} & \xi_2^{k-1} & \ldots & \xi_k^{k-1} \end{bmatrix},$$

which is a Vandermonde matrix. Since the values of $\xi_1, \xi_2, \ldots, \xi_k$ are distinct, K_0 is non-singular.

- Case 2: the characteristic polynomial (2.13) has $m < k$ distinct multiple roots $\xi_1, \xi_2, \ldots, \xi_m$. Clearly, since S has dimension k, the sequences $\{\xi_i^n\}_{n\in\mathbb{N}}$, $i=1, 2, \ldots, m$, are not enough to provide the needed set of k linearly independent solutions. The following result, whose proof is left to the reader, is useful to find k linearly independent solutions in this case.

Theorem 2.4 *If the characteristic polynomial (2.13) has $m < k$ distinct multiple roots $\xi_1, \xi_2, \ldots, \xi_m$, let us denote by μ_i the multiplicity of the ξ_i, $i = 1, 2, \ldots, m$. Then, the $k = \sum_{i=1}^{m} \mu_i$ sequences*

$$\begin{array}{llll} \{\xi_1^n\}_{n\in\mathbb{N}}, & \{n\xi_1^n\}_{n\in\mathbb{N}}, & \ldots, & \{n^{\mu_1-1}\xi_1^n\}_{n\in\mathbb{N}}, \\ \{\xi_2^n\}_{n\in\mathbb{N}}, & \{n\xi_2^n\}_{n\in\mathbb{N}}, & \ldots, & \{n^{\mu_2-1}\xi_2^n\}_{n\in\mathbb{N}}, \\ & & \vdots & \\ \{\xi_m^n\}_{n\in\mathbb{N}}, & \{n\xi_m^n\}_{n\in\mathbb{N}}, & \ldots, & \{n^{\mu_m-1}\xi_m^n\}_{n\in\mathbb{N}} \end{array}$$

are linearly independent solutions of (2.8).

- Case 3: the characteristic polynomial (2.13) has pairs of complex conjugate roots $\xi = \rho e^{i\vartheta}$ and $\overline{\xi} = \rho e^{-i\vartheta}$. Among the k linearly independent solutions, those corresponding to the complex conjugate roots are

$$\{\rho^n \sin(n\vartheta)\}_{n\in\mathbb{N}}, \qquad \{\rho^n \cos(n\vartheta)\}_{n\in\mathbb{N}}.$$

Let us now provide some examples of computation of solutions to both homogeneous and inhomogeneous scalar linear difference equations with constant coefficients.

Example 2.1 (Fibonacci Equation) Consider the following homogeneous difference equation

$$y_{n+2} - y_{n+1} - y_n = 0, \tag{2.14}$$

equipped by the initial values $y_0 = y_1 = 1$, defining the famous Fibonacci sequence. The corresponding characteristic polynomial is

$$x^2 - x - 1 = 0,$$

whose roots are the real distinct numbers

$$\frac{1+\sqrt{5}}{2}, \quad \frac{1-\sqrt{5}}{2}.$$

Then, the general solution of (2.14) is given by

$$y_n = \sigma_1 \left(\frac{1+\sqrt{5}}{2}\right)^n + \sigma_2 \left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \in \mathbb{N}.$$

The values of σ_1 and σ_2 can be computed by imposing the initial condition $y_0 = y_1 = 1$, obtaining

$$y_n = \left(\frac{\sqrt{5}+5}{10}\right)\left(\frac{1+\sqrt{5}}{2}\right)^n + \left(\frac{5-\sqrt{5}}{10}\right)\left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \in \mathbb{N}.$$

Example 2.2 Consider the following difference equation

$$y_{n+2} - y_{n+1} - y_n = 2^n, \tag{2.15}$$

equipped by the initial values $y_0 = y_1 = 1$. We have already computed the general solution of the homogeneous equation in Example 2.1 and, due to Theorem 2.3, we only need to find a particular solution $\overline{y}_{n\in\mathbb{N}}$ of (2.15). We make the ansatz

$$\overline{y}_n = a2^n, \qquad n \in \mathbb{N},$$

(continued)

Example 2.2 (continued)
i.e. we suppose that such a particular solution has a similar expression as in the right-hand side of (2.15). Replacing such an ansatz in (2.15) gives $a = 1$. Then, the general solution of (2.15) is

$$y_n = 2^n + \sigma_1 \left(\frac{1+\sqrt{5}}{2}\right)^n + \sigma_2 \left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \in \mathbb{N}.$$

The values of σ_1 and σ_2 are obtained by imposing the initial conditions $y_0 = y_1 = 1$, leading to

$$y_n = 2^n - \frac{\sqrt{5}}{5} \left(\frac{1+\sqrt{5}}{2}\right)^n + \frac{\sqrt{5}}{5} \left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \in \mathbb{N}.$$

Example 2.3 Consider the following difference equation

$$y_{n+6} - 6y_{n+5} + y_{n+4} + 28y_{n+3} - 72y_{n+2} + 208y_{n+1} - 240y_n = 0. \tag{2.16}$$

The corresponding characteristic polynomial (2.13) assumes the form

$$x^6 - 6x^5 + x^4 + 28x^3 - 72x^2 + 208x - 240 = 0$$

and its roots are 2, with multiplicity 2, $\pm 2i = 2e^{\pm i\pi/2}$, 5 and -3. Then, the general solution of (2.16) is given by

$$y_n = \sigma_1 2^n + \sigma_2 n 2^n + \sigma_3 2^n \sin\left(n\frac{\pi}{2}\right) + \sigma_4 2^n \cos\left(n\frac{\pi}{2}\right) + \sigma_5 5^n + \sigma_6 (-3)^n, \quad n \in \mathbb{N}.$$

2.3 Step-by-Step Schemes

As aforementioned, a difference equation is the discretized version of a differential equation; in other terms, numerical methods approximating the solutions of differential equations are given by difference equations that, in general, we are not able to solve, especially in the nonlinear case. However, difference equations are very important in order to give rise to *step-by-step* numerical schemes for the computation of approximate solutions of (1.1), as well clarified, for instance, in

monographs [18, 20, 62, 67, 161, 170, 172, 195, 198, 223, 228, 287, 319] and all references therein.

In order to understand the idea of step-by-step discretizations, let us introduce the simplest example of numerical method for (1.1), obtained by means of Taylor series arguments. Since the solution in t_0 is already given by the initial value of (1.1), the first approximate value we have to compute is $y_1 \approx y(t_1)$. If we assume that the solution of (1.1) is sufficiently smooth, we can expand $y(t_1) = y(t_0 + h)$ in Taylor series around t_0, obtaining

$$y(t_1) = y(t_0) + hy'(t_0) + \frac{h^2}{2}y''(t_0) + O(h^3).$$

Neglecting the terms from the second order on yields

$$y(t_1) \approx y(t_0) + hy'(t_0).$$

This is an approximate equality between exact values that can be regarded as an exact equality between approximate values, i.e.

$$y_1 = y_0 + hf(t_0, y_0). \tag{2.17}$$

This equality performs what is graphically displayed in Fig. 2.1, i.e., the computation of y_1 only requires the knowledge of y_0.

We can proceed in similar way for the computation of $y_2 \approx y(t_2)$, given y_1, via Taylor expansion of $y(t_1+h)$ around t_1 and truncating at the first order. We obtain an approximate equality between exact values leading to the following exact equality involving approximate values:

$$y_2 = y_1 + hf(t_1, y_1). \tag{2.18}$$

As visible from Eq. (2.18) and from its graphical description given in Fig. 2.2, the computation of y_2 only relies on the knowledge of y_1, already computed in the previous step (2.17).

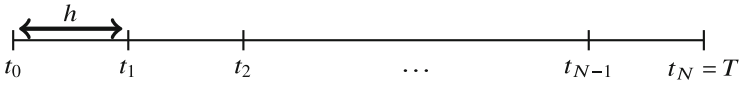


Fig. 2.1 Graphical description of the first step of Euler method, according to Eq. (2.17)

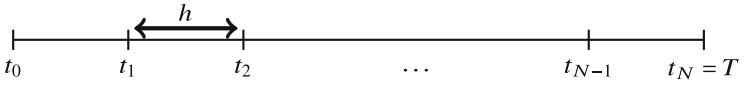


Fig. 2.2 Graphical description of the second step of Euler method, according to Eq. (2.18)

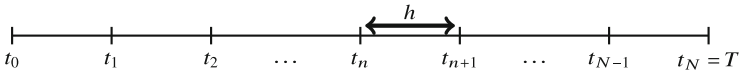


Fig. 2.3 Graphical description of the generic n-th step of Euler method, according to Eq. (2.19)

These arguments can clearly be generalized to perform the generic step from t_n to t_{n+1}, for the computation of $y_{n+1} \approx y(t_{n+1})$ given y_n. Indeed, expanding $y(t_n + h)$ in Taylor series around t_n leads to

$$y(t_n + h) = y(t_n) + hy'(t_n) + \frac{h^2}{2} y''(t_n) + O(h^3)$$

and neglecting the terms from the second order on yields

$$y(t_n + h) \approx y(t_n) + hy'(t_n).$$

This is an approximate equality between exact values that can be regarded as an exact equality between approximate values, i.e.,

$$y_{n+1} = y_n + hf(t_n, y_n). \tag{2.19}$$

Equation (2.19) is the well-known *Euler method* and its graphical description is given in Fig. 2.3. As we can appreciate from Eq. (2.19), the computation of the new approximate value y_{n+1} only relies on the knowledge of the approximate solution y_n referring to the previous point of the grid: in other terms, Euler method is a *one-step method*.

Euler method (2.19) is a first order nonlinear difference equation for which we are not able to a-priori provide a solution in closed form but, as previously stated, it can be used to activate the step-by-step scheme summarized as follows:

- y_0 is the initial value given by the continuous problem (1.1);
- the value of y_1 is computed from that of y_0 via Eq. (2.17);
- the value of y_1 permits the computation of y_2 via Eq. (2.18);
- in general, the computation of y_{n+1} relies on the knowledge of y_n, according to Eq. (2.19);
- this step-by-step process proceeds until the approximate solution in the last point t_N of the grid $\mathcal{I}_h$ is computed from that in t_{N-1} as

$$y_N = y_{N-1} + hf(t_{N-1}, y_{N-1}).$$

Euler method (2.19) is also an *explicit* method, i.e., the right-hand side of (2.19) does not depend on y_{n+1}, but it allows its direct computation in terms of y_n.

We now provide a simple Matlab implementation of Euler method (2.19) applied to (1.1), given in Program 2.1. In this program, the numerical solution is stored in a

matrix computed columnwise; its i-th column contains the approximate solution in the grid point $t_i \in \mathcal{I}_h$. The implementation is given with reference to the uniform grid (2.1). Let us observe that the function `f.m` required in Program 2.1 is reported in Appendix A, where a selection of test problems is proposed.

It is now worth highlighting the clear distinction between step-by-step and iterative methods. Indeed, step-by-step schemes do not provide any refinements to the solution of (1.1), as it happens in the case of iterative schemes. Step-by-step approximations with fixed stepsize compute the approximate solution in the grid points once, without refining the value of the solution until a prescribed tolerance is achieved, as it happens for iterative methods. Of course, it is also possible to refine the solution in a step-by-step scheme until a certain tolerance is reached: this is a more advanced topic, mostly based on error control in adaptive grids, that will be discussed in Chap. 7.

Program 2.1 (Euler Method)

```
% Function implementing Euler method on a uniform grid,
% for the numerical solution of a d-dimensional ODE.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
% - y0: initial value;
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: d×N matrix whose i-th column y(:,i) stores the
%      approximate value in the i-th grid point, i=1,2,...,N.

function [t,y]=euler(problem,tspan,y0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0);
y=zeros(d,N);
y(:,1)=y0+h*f(problem,tspan(1),y0);
for i=2:N
   y(:,i)=y(:,i-1)+h*f(problem,t(i-1),y(:,i-1));
end
```

2.4 A Theory of One-Step Methods

This section is devoted to introducing three basic accuracy and stability concepts necessary in any numerical discretization of (1.1), i.e., consistency, zero-stability

and convergence. It is important to stress their importance from the very beginning: nothing less than consistency, zero-stability and convergence can be admitted in any numerical method for ODEs. Indeed, they provide three fundamental ingredients creating a first meaningful bridge between the continuous problem (1.1) and its discretization, as it will be clarified in the following pages for a reference family of methods.

In particular, let us consider as case study the following family of explicit one-step methods

$$y_{n+1} = y_n + h\varphi(t_n, y_n; h), \tag{2.20}$$

where the incremental function $\varphi : [t_0, T] \times \mathbb{R}^d \times [0, +\infty) \to \mathbb{R}^d$ characterizes the method itself. For instance, Euler method (2.19) is recovered if

$$\varphi(t_n, y_n; h) = f(t_n, y_n).$$

2.4.1 Consistency

We focus our attention on a single step of method (2.20), from the grid point t_n to t_{n+1}, which can be assumed as a discrete counterpart of the following *local problem*

$$\begin{aligned} u'(t) &= f(t, u(t)), \quad t \in [t_n, t_{n+1}], \\ u(t_n) &= y(t_n), \end{aligned} \tag{2.21}$$

where $u(t)$ is the restriction of $y(t)$ when t belongs to the interval $[t_n, t_{n+1}]$. Applying (2.20) to the local problem (2.21) implies the tacit assumption that the initial value is not affected by any source of error: indeed, the initial value of the local problem (2.21) is assumed to be equal to the exact value $y(t_n)$ of the global problem (1.1). This tacit assumption is the so-called *localizing assumption*: in other terms, under the localizing assumption, we study the application of a numerical method over a single step from t_n to t_{n+1}, with the hypothesis the value in t_n is exact. Clearly, by applying (2.20) to (2.21), we obtain y_{n+1} as an approximation of $u(t_{n+1})$.

Under the localizing assumption, method (2.20) assumes the form

$$y_{n+1} = y(t_n) + h\varphi(t_n, y(t_n); h)$$

or, alternatively,

$$\frac{y_{n+1} - y(t_n)}{h} - \varphi(t_n, y(t_n); h) = 0$$

If we replace y_{n+1} by its exact counterpart $u(t_{n+1})$, we obtain a measure of the residuum

$$T(t_n, y(t_n); h) = \frac{u(t_{n+1}) - y(t_n)}{h} - \varphi(t_n, y(t_n); h), \tag{2.22}$$

which is the so-called *local truncation error*, measuring the gap between the exact scaled increment $\frac{1}{h}(u(t_{n+1}) - y(t_n))$ and the numerical one $\varphi(t_n, y(t_n); h)$, under the localizing assumption. Clearly, (2.22) is equivalent to

$$T(t_n, y(t_n); h) = \frac{1}{h}(u(t_{n+1}) - y_{n+1}),$$

that provides a measure of the error intended as difference between the exact and the approximate solution, under the localizing assumption. Let us now provide the following definition.

Definition 2.4 A given one-step method (2.20) is *consistent* if, for any $(t, y) \in [t_0, T] \times \mathbb{R}^d$,

$$\lim_{h \to 0} T(t, y; h) = 0.$$

In other terms, consistent one-step methods (2.20) satisfy

$$\varphi(t, y; 0) = f(t, y), \tag{2.23}$$

for any $(t, y) \in [t_0, T] \times \mathbb{R}^d$. We can thus say that consistency is the coherence between the numerical increment of (2.20) and the vector field of (1.1), as h tends to 0. Clearly, due to (2.23), Euler method is a consistent method.

Let us stress again that consistency focuses on a local analysis of numerical methods, strongly relying on the localizing assumption. On the other hand, next sections are focused on two global notions (zero-stability and convergence) that neglect the unrealistic localizing assumption and consider the error that cumulates overall the step-by-step process.

Let us now give the following definition, introducing an accuracy measure for one-step methods (2.20).

Definition 2.5 A one-step method (2.20) has *order* p if, for a chosen vector norm $\|\cdot\|$, there exists a real constant $C > 0$ such that

$$\|T(t, y; h)\| \leq Ch^p,$$

for any $(t, y) \in [t_0, T] \times \mathbb{R}^d$, where C is independent on t, y and h.

Hence, an order p method (2.20) satisfies

$$T(t, y; h) = O(h^p)$$

and, as a consequence, a consistent method has order $p \geq 1$. The local truncation error of an order p method is then of the type

$$T(t, y; h) = \tau(t, y)h^p + O(h^{p+1}),$$

where the coefficient of the leading error term $\tau(t, y)$ is called *principal error function*.

Example 2.4 Let us compute the order of Euler method (2.19), whose local truncation error (2.22) is given by

$$T(t_n, y(t_n); h) = \frac{u(t_{n+1}) - y(t_n)}{h} - f(t_n, y(t_n)).$$

Taking into account the localizing assumption $u(t_n) = y(t_n)$ in (2.21), we have

$$T(t_n, y(t_n); h) = \frac{u(t_{n+1}) - u(t_n)}{h} - u'(t_n).$$

Finally, applying Taylor formula yields

$$T(t_n, y(t_n); h) = \frac{1}{h}\left(u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(\xi) - u(t_n)\right) - u'(t_n) = \frac{1}{2}hu''(\xi),$$

with $\xi \in (t_n, t_{n+1})$. Since $u'(t) = f(t, u(t))$, then

$$u''(t) = f_t(t, u(t)) + f_y(t, u(t))f(t, u(t)).$$

(continued)

Example 2.4 (continued)
If the vector field f and its first derivates are uniformly bounded in $[t_n, t_{n+1}]$ by $2C$, then

$$\|T(t_n, y_n; h)\| \leq Ch,$$

i.e., Euler method has order 1.

2.4.2 Zero-Stability

It is now worth introducing the following operators.

Definition 2.6 For a given function of class $C^1([t_0, T])$

$$v : [t_0, T] \to \mathbb{R}^d,$$

we define the *residual operator* associated to (1.1) as

$$R(v) = v'(t) - f(t, v(t)). \tag{2.24}$$

Definition 2.7 For a given grid function

$$v : \mathcal{I}_h \to \mathbb{R}^d,$$

let us denote by v_n its value in $t_n \in \mathcal{I}_h$. We define the *numerical residual operator* in t_n, associated to (2.20), as

$$R_h(v_n) = \frac{v_{n+1} - v_n}{h} - \varphi(t_n, v_n; h),$$

for $n = 0, 1, \ldots, N-1$.

When the residual operators are respectively evaluated in the exact solution of (1.1) and its approximation computed by (2.20), we have

$$R(y) = 0, \quad R_h(y_n) = 0.$$

Moreover, evaluating the numerical residual operator R_h in the exact solution $y(t_n)$ recovers the local truncation error (2.22), since

$$R_h(y(t_n)) = \frac{y(t_{n+1}) - y(t_n)}{h} - \varphi(t_n, y(t_n); h) = T(t_n, y(t_n); h).$$

We aim to provide the numerical counterpart of the continuous dependence on the initial value and the vector field, described in Theorem 1.4. In other terms, we aim to analyze the sensitivity of one-step methods (2.20) with respect to the effect of perturbations on the initial data and the vector fields of the problem. To this purpose, we introduce the vector

$$\mathbf{y}_h = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_N \end{bmatrix} \tag{2.25}$$

collecting the numerical approximations of the solution to the original problem (1.1) in each grid point, obtained by (2.20). We also introduce the vector

$$\widetilde{\mathbf{y}}_h = \begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \\ \vdots \\ \widetilde{y}_N \end{bmatrix} \tag{2.26}$$

of the numerical approximations of the solution to the perturbed problem (1.12), obtained by (2.20). We also denote by $\delta = y_0 - \widetilde{y}_0$ the difference between the initial values of the original problem (1.1) and the perturbed one (1.12).

Clearly, $R_h(y_n) = 0$, while we suppose that $R_h(\widetilde{y}_n) = \varepsilon_n$ and collect all these values in the vector

$$\varepsilon = \begin{bmatrix} \varepsilon_0 \\ \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{bmatrix}.$$

Then, we give the following definition.

Definition 2.8 A one-step method (2.20) is *zero-stable* if there exists $K > 0$ such that, for any value of the stepsize $h \in [0, h_0]$, with $h_0 > 0$, the following inequality holds true

$$\|\mathbf{y}_h - \widetilde{\mathbf{y}}_h\|_\infty \leq K(\|\delta\|_\infty + \|\varepsilon\|_\infty). \tag{2.27}$$

We observe that the definition of zero-stability is free from any localizing assumption. It states that, for bounded perturbations of the initial value and the vector field of the problem, the perturbation on the numerical solution remains bounded as well, as far as the stepsize h is in a neighborhood of 0, hence for small values of h. Zero-stability inequality (2.27) given in Definition 2.8 is not intended to provide a quantitative measure of the gap between $\mathbf{y}_h$ and $\widetilde{\mathbf{y}}_h$; actually, we even do not know much on the order of magnitude of the perturbation on the solutions: indeed, we only know that the mentioned gap does not blow-up when h is very small.

Lipschitz continuity of the vector field is enough for the continuous dependence of the exact solution on the initial value and the vector field, as proved in Theorem 1.4. We now aim to prove that a similar condition on the incremental function φ in (2.20) is enough to guarantee zero-stability. To this purpose, we need to prove the following discrete Grönwall lemma.

Lemma 2.2 (Discrete Grönwall Lemma) *Suppose that $e_0, e_1, \ldots, e_N$ are real numbers satisfying the following inequality*

$$e_{n+1} \leq a_n e_n + b_n, \quad n = 0, 1, \ldots, N-1,$$

with $a_n > 0$, $b_n \in \mathbb{R}$. Then,

$$e_n \leq \left(\prod_{k=0}^{n-1} a_k\right) e_0 + \sum_{k=0}^{n-1} \left(\prod_{\ell=k+1}^{n-1} a_\ell\right) b_k, \quad n = 0, 1, \ldots, N. \tag{2.28}$$

Proof Let us denote the right-hand side of (2.28) by

$$E_n = \left(\prod_{k=0}^{n-1} a_k\right) e_0 + \sum_{k=0}^{n-1} \left(\prod_{\ell=k+1}^{n-1} a_\ell\right) b_k,$$

with $E_0 = e_0$. Let us isolate the terms corresponding to the index $n-1$, obtaining

$$E_n = a_{n-1}\left(\prod_{k=0}^{n-2} a_k\right) e_0 + \sum_{k=0}^{n-2} a_{n-1}\left(\prod_{\ell=k+1}^{n-2} a_\ell\right) b_k + b_{n-1},$$

i.e.,

$$E_n = a_{n-1}E_{n-1} + b_{n-1}.$$

Then,

$$e_n - E_n \le a_{n-1}e_{n-1} + b_{n-1} - (a_{n-1}E_{n-1} + b_{n-1}) = a_{n-1}(e_{n-1} - E_{n-1}).$$

Let us now proceed by induction on n to prove that $e_n \le E_n$, for any n. For $n = 1$ we have

$$e_1 - E_1 \le a_0(e_0 - E_0) = 0.$$

Suppose that $e_{n-1} \le E_{n-1}$. Then,

$$e_n - E_n \le a_{n-1}(e_{n-1} - E_{n-1}) \le 0,$$

that gives the thesis. □

Theorem 2.5 *A one-step method (2.20) is zero-stable if there exists $M > 0$ such that*

$$\|\varphi(t, v, h) - \varphi(t, w, h)\| \le M\|v - w\|, \tag{2.29}$$

for any $(t, v, h), (t, w, h) \in [t_0, T] \times \mathbb{R}^d \times [0, h_0]$.

Proof Consider the vectors of numerical solutions (2.25) and (2.26), whose generic n-th entry is given by

$$y_{n+1} = y_n + h\varphi(t_n, y_n; h),$$

$$\widetilde{y}_{n+1} = \widetilde{y}_n + h\varphi(t_n, \widetilde{y}_n; h) + h\varepsilon_n.$$

Side-by-side subtracting, passing to the norms and applying (2.29) yields

$$\|y_{n+1} - \widetilde{y}_{n+1}\| \le (1 + hM)\|y_n - \widetilde{y}_n\| + h\|\varepsilon\|_\infty.$$

Denoting by

$$a = 1 + hM, \quad e_n = \|y_n - \widetilde{y}_n\|, \quad b = h\|\varepsilon\|_\infty,$$

last inequality is equivalent to

$$e_{n+1} \leq a e_n + b$$

and applying Lemma 2.2, we have

$$e_n \leq \left(\prod_{k=0}^{n-1} a\right) e_0 + b \sum_{k=0}^{n-1} \left(\prod_{\ell=k+1}^{n-1} a\right).$$

Clearly,

$$\prod_{k=0}^{n-1} a \leq \prod_{k=0}^{N-1} (1 + hM) \leq \prod_{k=0}^{N-1} \mathrm{e}^{hM} = \mathrm{e}^{NhM} = \mathrm{e}^{(T-t_0)M}$$

and, analogously,

$$\prod_{\ell=k+1}^{n-1} a \leq \prod_{k=0}^{N-1} (1 + hM) \leq \mathrm{e}^{(T-t_0)M}.$$

We finally obtain

$$e_n \leq \mathrm{e}^{(T-t_0)M} (e_0 + (T - t_0)\|\varepsilon\|_\infty),$$

i.e.,

$$\|\mathbf{y}_h - \widetilde{\mathbf{y}}_h\|_\infty \leq \mathrm{e}^{(T-t_0)M} \left(\|\delta\|_\infty - (T - t_0)\|\varepsilon\|_\infty\right).$$

The thesis holds true with $K = \mathrm{e}^{(T-t_0)M} \max\{1, T - t_0\}$. □

Clearly, according to Theorem 2.5, Euler method (2.19) is zero-stable, since the numerical increment inherits its Lipschitz continuity from that of the vector field of the problem (1.1), which is always supposed to be Hadamard well-posed.

2.4.3 Convergence

Last point of this section is focused on understanding what happens to the accuracy of the numerical scheme if we reduce the stepsize or, equivalently, if we increase the

number of grid points. By zero-stability, the error intended as difference between the exact solution of the problem and its numerical approximation does not blow-up when the stepsize tends to zero. Through the notion of *convergence* we now introduce, one can also reinforce the stable behavior by stating that the error itself tends to 0, according to the following definition.

Definition 2.9 A one-step method (2.20) is *convergent* if

$$\lim_{h\to 0} \|\boldsymbol{v}_h - \mathbf{y}_h\|_\infty = 0, \tag{2.30}$$

where

$$\boldsymbol{v}_h = \begin{bmatrix} y_0 \\ y(t_1) \\ \vdots \\ y(t_N) \end{bmatrix}$$

and the vector norm $\|\cdot\|_\infty$ in Eq. (2.30) is defined as

$$\|v\|_\infty = \max_{0\le k\le n} |v_k|, \quad v \in \mathbb{R}^n.$$

The following important result allows to study the convergence of a numerical method in terms of consistency and zero-stability.

Theorem 2.6 *A consistent and zero-stable method* (2.20) *is convergent.*

Proof Let us apply the zero-stability inequality (2.27) to provide an estimate of $\|\boldsymbol{v}_h - \mathbf{y}_h\|_\infty$, obtaining

$$\|\boldsymbol{v}_h - \mathbf{y}_h\|_\infty \le K\|\varepsilon\|_\infty,$$

where

$$\varepsilon = \begin{bmatrix} R_h(y_0) \\ R_h(y(t_1)) \\ \vdots \\ R_h(y(t_N)) \end{bmatrix}.$$

Table 2.1 Example 2.25: an experimental confirmation of the convergence of Euler method, applied to the linear scalar problem (2.31)

h	$\lvert y(10) - y_{\text{EUL}}\rvert$
0.1	$1.86 \cdot 10^{-9}$
0.01	$3.78 \cdot 10^{-10}$
0.001	$4.09 \cdot 10^{-11}$
0.0001	$4.12 \cdot 10^{-12}$

As previously mentioned, the operator R_h evaluated in the exact solution gives the local truncation error, that tends to 0 by consistency, leading to the thesis. □

The vice versa is also true; we will prove this result in Chap. 3, in a more general setting. By Theorem 2.6, we have also proved that Euler method is convergent. Let us check this property also experimentally, thanks to the following numerical test obtained by applying Program 2.1 applied to a linear scalar equation.

Example 2.25 We aim to experimentally check the convergence of Euler method (2.19), through its application to the following linear scalar problem

$$\begin{aligned} y'(t) &= -2y(t), \quad t \in [0, 10], \\ y(0) &= 1, \end{aligned} \tag{2.31}$$

whose exact solution is $y(t) = \mathrm{e}^{-2t}$. We list in Table 2.1 the values of the difference $|y(10) - y_{\text{EUL}}|$, where $y_{\text{EUL}} \approx y(10)$ is computed by Euler method (2.19). As visible from the table, the more the value of the stepsize diminishes, the more the difference between the numerical and the exact solutions becomes smaller. This behavior is in agreement with the proved convergence of Euler method.

2.5 Handling Implicitness

Euler method (2.19) is a very basic scheme, achieving the lowest admissible order for a convergent method (i.e., $p = 1$) and, as we will see in remainder, very poor in terms of stability and conservation properties. As already noted, this method is explicit, then it is very easy to implement. This scheme also admits an *implicit* version, obtainable by expanding $y(t_n)$ in Taylor series around $t_n + h$ (again, assuming that $y(t)$ is sufficiently regular), i.e.,

$$y(t_n) = y(t_n + h) - hy'(t_n + h) + O(h^2).$$

Neglecting the terms from the second order on leads to

$$y(t_n + h) \approx y(t_n) + hy'(t_n + h).$$

This is an approximate equality between exact values that can be regarded as an exact equality between approximate values, i.e.,

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}). \tag{2.32}$$

Equation (2.32) gives the so-called *implicit Euler method*. If f is a nonlinear function, (2.32) is a nonlinear algebraic equation to be solved at each step, in order to compute y_{n+1} from y_n, for any $n \geq 0$. This issue certainly heightens the required computational effort of (2.32) in comparison with its explicit version (2.19), but sometimes it is unavoidable in order to achieve desirable properties, as it will be clarified in the following chapters.

The reader can prove that also the implicit Euler method has order 1 (see Exercise 4 in Sect. 2.6), through arguments analogous to those used in Example 2.4. Now we can understand that both versions of Euler method (2.19) and (2.32) have the minimum acceptable order of convergence, that is equal to 1. In order to achieve higher accuracy, the structure of the numerical method should be enriched a bit more.

We show a very well known example of order 2 method. So far, we have used Taylor series expansions as a constructive technique; next example shows the development of a method by means of another powerful tool, i.e., numerical quadrature.

Example 2.26 (Trapezoidal Method) Let us consider, for $t \geq t_n$, the integral form of (1.1), i.e.,

$$y(t) = y(t_n) + \int_{t_n}^{t} f(s, y(s))\, \mathrm{d}s$$

and evaluating in t_{n+1} yields

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(s, y(s))\mathrm{d}s.$$

The integral in the right-hand side of last equation can be approximated by means of a chosen numerical quadrature formula. For instance, let us apply the trapezoidal quadrature rule

$$\int_{t_n}^{t_{n+1}} f(s, y(s))\mathrm{d}s \approx \frac{h}{2}\left(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))\right),$$

(continued)

Example 2.26 (continued)
leading to

$$y(t_{n+1}) \approx y(t_n) + \frac{h}{2}\left(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))\right).$$

This is an approximate equality involving exact values that can be regarded as an exact equality involving approximate values, i.e.,

$$y_{n+1} = y_n + \frac{h}{2}\left(f(t_n, y_n) + f(t_{n+1}, y_{n+1})\right). \tag{2.33}$$

Equation (2.33) is the so-called *trapezoidal method*. Let us analyze its order of convergence, by inspecting the local truncation error (2.22)

$$T(t_n, y(t_n); h) = \frac{u(t_{n+1}) - y(t_n)}{h} - \frac{1}{2}\left(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))\right).$$

Taking into account the localizing assumption $u(t_n) = y(t_n)$ in (2.21) and by means of Taylor series expansions, we have

$$\begin{aligned} T(t_n, y(t_n); h) &= \frac{u(t_{n+1}) - u(t_n)}{h} - \frac{1}{2}\left(u'(t_n) + u'(t_{n+1})\right) \\ &= \frac{1}{h}\left(u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(t_n) + \frac{h^3}{6}u(t_n) - u(t_n)\right) \\ &= -\frac{1}{12}h^2\, u'''(t_n) + O(h^3). \end{aligned}$$

Hence, the trapezoidal method has order 2.

Both the implicit Euler and trapezoidal methods are implicit schemes. As a consequence, if the vector field of (1.1) is nonlinear, the computation of the numerical solution requires the solution of nonlinear systems of algebraic equations at each step. To this purpose, fixed point iterations may be used. For a given nonlinear system of algebraic equations $z = g(z)$, with $g : \mathbb{R}^d \to \mathbb{R}^d$, fixed point iterations require the choice of an arbitrary initial guess $z^{[0]} \in \mathbb{R}^d$ and proceed according to the following scheme

$$z^{[\nu+1]} = g(z^{[\nu]}), \quad \nu \geq 0, \tag{2.34}$$

whose convergence is object of the following well-known theorem.

Theorem 2.7 *For a chosen vector norm $\|\cdot\|$, let $g : \mathbb{R}^d \to \mathbb{R}^d$ satisfy the Lipschitz condition*

$$\|g(y) - g(y^*)\| \leq L\|y - y^*\|, \quad y, y^* \in \mathbb{R}^d.$$

If $0 \leq L < 1$, there exists a unique $\alpha \in \mathbb{R}^d$ satisfying $\alpha = g(\alpha)$ and such that, for any arbitrary initial guess $z^{[0]} \in \mathbb{R}^d$, the iterative scheme (2.34) converges to α, i.e.

$$\lim_{\nu \to \infty} \|z^{[\nu]} - \alpha\| = 0.$$

The proof of Theorem 2.7 is here omitted, but the reader can be find it in monographs on numerical methods, e.g., [170, 292, 329]. We also remark that the convergence of the fixed point iterations for implicit numerical methods approximating the solution of initial value problems (1.1) will be studied in Chap. 3 in a general setting (including the implicit Euler and trapezoidal methods as special cases).

Let us now summarize the numerical scheme relying on the trapezoidal method (2.33) and handled by fixed point iterations: in order to advance from t_n to t_{n+1},

- we arbitrarily choose an initial guess $y_{n+1}^{[0]} \in \mathbb{R}^d$. To speed up the convergence of the iterative process, a smart choice may be $y_{n+1}^{[0]} = y_n$;
- we perform fixed point iterations

$$y_{n+1}^{[\nu+1]} = y_n + \frac{h}{2}\left(f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{[\nu]})\right), \quad \nu \geq 0,$$

stopping at the iteration M if

$$\|y_{n+1}^{[M]} - y_{n+1}^{[M-1]}\| \leq tol,$$

being tol an a-priori prescribed accuracy. Then $y_{n+1} = y_{n+1}^{[M]}$.

Program 2.2 shows a Matlab implementation of this scheme with $tol = 10^{-15}$. Let us provide an example of use of this program for the numerical solution of (2.31).

Table 2.2 Comparison between the explicit Euler (2.19) and trapezoidal (2.33) methods, applied to Eq. (2.31)

h	$\lvert y(10) - y_{\text{EUL}}\rvert$	$\lvert y(10) - y_{\text{TRAP}}\rvert$
0.1	$1.86 \cdot 10^{-9}$	$1.34 \cdot 10^{-10}$
0.01	$3.78 \cdot 10^{-10}$	$1.37 \cdot 10^{-12}$
0.001	$4.09 \cdot 10^{-11}$	$1.37 \cdot 10^{-14}$
0.0001	$4.12 \cdot 10^{-12}$	$7.17 \cdot 10^{-17}$

Example 2.27 Carrying on the analysis provided in Example 2.25, we now use Program 2.2 to solve Eq. (2.31) with the trapezoidal method (2.33) and, finally, compare the performances of the explicit Euler and trapezoidal methods. Table 2.2, enriching the results already displayed in Table 2.1, shows the values of the difference $|y(10) - y_{\text{TRAP}}|$, where $y_{\text{TRAP}} \approx y(10)$ is computed by the trapezoidal method. Since the trapezoidal method has higher order, employing the same stepsize, its associated error is smaller and diminishes much faster than that provided by the explicit Euler method.

Program 2.2 (Trapezoidal Method)

```
% Function implementing the trapezoidal method on a uniform
% grid, for the numerical solution of a d-dimensional ODE.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
% - y0: initial value;
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: d×N matrix whose i-th column y(:,i) stores the
%      approximate value in the i-th grid point, i=1,2,...,N.

function [t,y]=trapezoidal(problem,tspan,y0,h)
tol=1e-15;
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0);
y=zeros(d,N);
f0=f(problem,tspan(1),y0);
yold=y0;
ynew=y0+h*(f0+f(problem,t(1),yold))/2;
```

(continued)

Program 2.2 (continued)

```
while norm(ynew-yold,'inf')>tol
      yold=ynew;
      ynew=y0+h*(f0+f(problem,t(1),yold))/2;
end
y(:,1)=ynew;
for i=2:N
    fPrev=f(problem,t(i-1),y(:,i-1));
    yold=y(:,i-1);
    ynew=y(:,i-1)+h*(fPrev+f(problem,t(i),yold))/2;
    while norm(ynew-yold,'inf')>tol
          yold=ynew;
          ynew=y(:,i-1)+h*(fPrev+f(problem,t(i),yold))/2;
    end
    y(:,i)=ynew;
end
```

2.6 Exercises

1. Solve the following scalar linear difference equations:

   (a) $y_{n+2} - 2y_{n+1} - y_n = 3^n$,
   (b) $y_{n+4} - 2y_{n+3} - y_{n+2} + y_n = 0$,
   (c) $y_{n+5} + 2y_{n+4} + 2y_{n+3} + 2y_{n+2} + y_{n+1} = 2^n$.

2. Using the hypothesis of convergence of Euler method (2.19), recover the expression of the original differential problem, as h tends to 0.
   (Hint: it may be useful to write the method in the form $(y_{n+1} - y_n)/h = f(t_n, y_n)$ *and analyze what happens as* h *goes to 0).*
3. Prove that a numerical method for

$$y(t) = y(t_n) + \int_{t_n}^{t} f(s, y(s))\mathrm{d}s$$

   obtained by means of a quadrature rule for the approximation of the integral in the right-hand side inherits the same order of convergence of the underlying quadrature formula.
4. Prove the implicit Euler method (2.32) has order 1. Also give a proof of its convergence, exploiting Theorem 2.6.
5. Write a code in the programming language you prefer that computes the solution of (1.1) by the implicit Euler method (2.32) and provides a pointwise

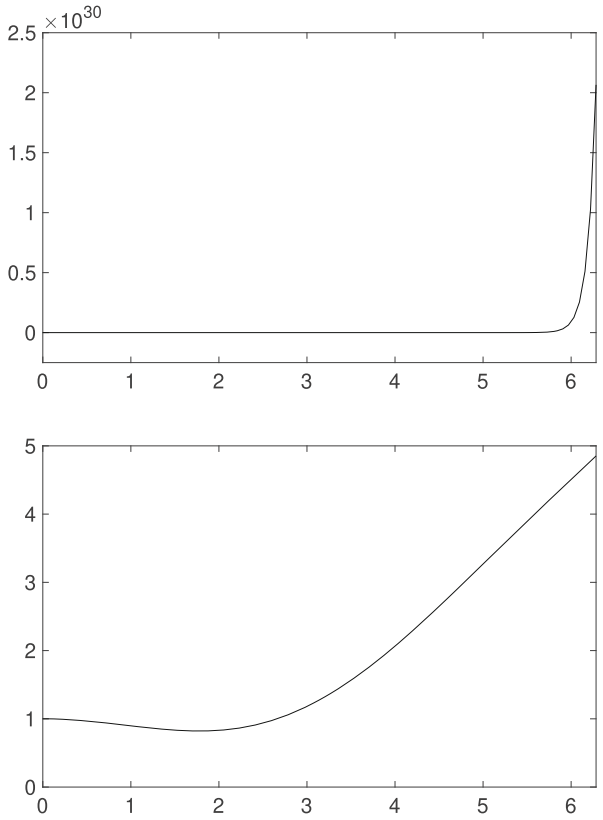


Fig. 2.4 Patterns of numerical solutions computed by two non-convergent methods (see Exercise 7)

error estimate. The algorithm behind a single step from t_n to t_{n+1} should be as follows:

- compute the numerical solution $y_{n+1} \approx y(t_{n+1})$ by applying (2.32) with a chosen stepsize h;
- compute another approximation $\widetilde{y}_{n+1} \approx y(t_{n+1})$ with two steps of (2.32) of length $h/2$;
- estimate the error as $|y_{n+1} - \widetilde{y}_{n+1}|$.

6. Rewrite Program (2.2) by replacing fixed point iterations with Newton iterations to handle the implicitness of the trapezoidal method.
7. Figure 2.4 shows the patterns of the numerical solutions in $[0, 2\pi]$ of a problem whose exact solution is $y(t) = \cos(t)$, computed by a couple of two different non-convergent one-step methods. Describe the reasons why each method is not convergent.

8. Given the three-parameter family of one-step methods

$$\alpha y_{n+1} = \beta y_n + h\gamma f(t_n, y_n), \quad \alpha, \beta, \gamma \in \mathbb{R},$$

(a) find the values of α, β and γ ensuring consistency;
(b) is order 2 achievable in correspondence of specific values of α, β and γ?

9. Discuss the zero-stability of the following one-parameter family of one-step methods

$$\alpha y_{n+1} = 2y_n + hf(t_n, y_n), \quad \alpha \in \mathbb{R}.$$

10. Analyze the convergence of the scheme

$$y_{n+1} = y_n + \frac{h}{2}\left(f(t_n, y_n) + f(t_{n+1}, \tilde{y}_{n+1})\right),$$

with

$$\tilde{y}_{n+1} = y_n + hf(t_n, y_n).$$

A single step of the overall scheme consists in two parts: a prediction of the value of the approximate solution in t_{n+1} by means of the explicit Euler method; a correction of this value via the trapezoidal method. Does the presence of the prediction step affect the second order of the trapezoidal method?

Chapter 3
Linear Multistep Methods

[. . .] this approach, which was first adopted by Dahlquist, leads to a mathematically well-rounded theory. It also leads to the discovery of new integration formulas which could not be obtained by the heuristic methods.

(Peter Henrici [206]. This quotation has also been highlighted by Ernst Hairer in [188])

One-step methods have been introduced, analyzed and implemented in Chap. 2. Even if they provide the simplest and maybe most intuitive family of step-by-step numerical schemes, enlarging this class with more complex methods could be useful in order to achieve better accuracy and stability properties. For this reason, we present a more general family of methods relying on a multistep structure, defined as follows.

3.1 The Principle of Multistep Numerical Integration

Definition 3.1 The family of *linear multistep methods* (LMMs), with respect to the discretization (2.1), is defined by the order k difference equation

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j}, \tag{3.1}$$

where $f_{n+j} = f(t_{n+j}, y_{n+j})$, $j = 0, 1, \ldots, k$, $n = 0, 1, \ldots, N$.

The integer k is usually denoted as the number of *steps* of the method and represents the order of the difference equation defining (3.1). Normally, we assume $\alpha_k = 1$ (as

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_3

usual, if this is not the case, all coefficients can be normalized in order to fall in this instance) and, moreover, $|\alpha_0| + |\beta_0| \neq 0$ in order to avoid α_0 and β_0 simultaneously equal to zero.

The family of LMMs includes all the one-step methods (hence, $k = 1$) introduced in Chap. 2, namely

- explicit Euler method (2.19), assuming that $\alpha_0 = -1, \alpha_1 = 1, \beta_0 = 1, \beta_1 = 0$;
- implicit Euler method (2.32), for $\alpha_0 = -1, \alpha_1 = 1, \beta_0 = 0, \beta_1 = 1$;
- the trapezoidal method (2.33), imposing $\alpha_0 = -1, \alpha_1 = 1, \beta_0 = \frac{1}{2}, \beta_1 = \frac{1}{2}$.

Let us now provide an example of LMM (3.1) depending on more than one step, obtained by means of proper numerical quadrature.

Example 3.1 We now aim to derive an example of LMM with $k = 2$, i.e., a two-step method. As in the construction of the trapezoidal method (2.33), let us consider, for $t \geq t_n$, the integral form of (1.1), i.e.,

$$y(t) = y(t_n) + \int_{t_n}^{t} f(s, y(s))\mathrm{d}s,$$

and evaluate it in t_{n+2}, obtaining

$$y(t_{n+2}) = y(t_n) + \int_{t_n}^{t_{n+2}} f(s, y(s))\mathrm{d}s.$$

Approximating the integral in the right-hand side by the Cavalieri-Simpson formula

$$\int_{t_n}^{t_{n+2}} f(s, y(s))\mathrm{d}s \approx \frac{h}{3}\left(f(t_n, y(t_n)) + 4f(t_{n+1}, y(t_{n+1})) + f(t_{n+2}, y(t_{n+2}))\right)$$

yields

$$y(t_{n+2}) \approx y(t_n) + \frac{h}{3}\left(f(t_n, y(t_n)) + 4f(t_{n+1}, y(t_{n+1})) + f(t_{n+2}, y(t_{n+2}))\right).$$

This is an approximate equality involving exact values that can be regarded as an exact equality involving approximate values, i.e.,

$$y_{n+2} = y_n + \frac{h}{3}\left(f_n + 4f_{n+1} + f_{n+2}\right), \tag{3.2}$$

that is the so-called *Milne-Simpson* method.

Differently from one-step methods, LMMs are not self-starting (unless $k = 1$). For instance, consider the Milne-Simpson method (3.2) when $n = 0$, leading to

$$y_2 = y_0 + \frac{h}{3}(f_0 + 4f_1 + f_2).$$

The value of y_0 is initial value given by the problem (1.1), but the value of y_1 is missing and it needs to be recovered in order to compute y_2 and launch the step-by-step procedure.

More in general, for the family of k-step methods (3.1) a proper starting method is needed to reconstruct the missing starting values $y_1, y_2, \ldots, y_{k-1}$. Such values can be recovered, for instance, by a suitable one-step method. Just as an example, if we employ Euler method (2.19) as starting method, the step-by-step numerical scheme described by (3.1) can be summarized as follows:

- y_0 is given by the initial value problem (1.1);
- compute the missing starting values $y_1, y_2, \ldots, y_{k-1}$ by repeatedly applying (2.19), i.e.,

$$\begin{aligned} y_1 &= y_0 + hf_0, \\ y_2 &= y_1 + hf_1, \\ &\vdots \\ y_{k-1} &= y_{k-2} + hf_{k-2}; \end{aligned}$$

- compute y_k by applying the LMM (3.1)

$$y_k + \sum_{j=0}^{k-1} \alpha_j y_j = h\beta_k f_k + \sum_{j=0}^{k-1} \beta_j f_j.$$

  We observe that, if $\beta_k \neq 0$, this step is equivalent to solving a nonlinear system of algebraic equations in y_k;
- go on applying (3.1) up to the computation of

$$y_N + \sum_{j=0}^{k-1} \alpha_j y_{N-k+j} = h\beta_k f_N + \sum_{j=0}^{k-1} \beta_j f_{N-k+j}.$$

Relevant examples of LMMs (3.1) can be computed, for instance, via polynomial interpolation, where the interpolation points are normally chosen among the grid points. Let us illustrate this idea through the following examples.

Example 3.2 (An Adams-Bashforth Method) Let us consider the integral formulation of (1.1)

$$y(t) = y(t_{n+1}) + \int_{t_{n+1}}^{t} f(s, y(s))\mathrm{d}s,$$

and evaluate it in t_{n+2}, obtaining

$$y(t_{n+2}) = y(t_{n+1}) + \int_{t_{n+1}}^{t_{n+2}} f(s, y(s))\mathrm{d}s.$$

Let us approximate $f(s, y(s))$ through the linear interpolant with respect to the nodes $(t_n, y(t_n))$ and $(t_{n+1}, y(t_{n+1}))$, leading to

$$f(s, y(s)) \approx f(t_n, y(t_n))\frac{s - t_{n+1}}{t_n - t_{n+1}} + f(t_{n+1}, y(t_{n+1}))\frac{s - t_n}{t_{n+1} - t_n}.$$

Hence,

$$\int_{t_n}^{t_{n+2}} f(s, y(s))\mathrm{d}s \approx -\frac{h}{2}\left(f(t_n, y(t_n)) - 3f(t_{n+1}, y(t_{n+1}))\right),$$

leading to

$$y_{n+2} = y_{n+1} - \frac{h}{2}\left(f_n - 3f_{n+1}\right), \tag{3.3}$$

that is the so-called *two-step Adams-Bashforth method*, which is an explicit method.

More in general, Adams-Bashforth methods are obtained by replacing the interpolating polynomial approximating f on a given set of nodes chosen among the grid points, excluding the point related to the advancing term, i.e., t_{n+k}. This choice leads to a family of explicit methods. Including t_{n+k} in the set of interpolation points leads to a family of implicit methods, the so-called *Adams-Moulton* formulae. An example is given below.

Example 3.3 (An Adams-Moulton Method) Let us consider again the integral form of (1.1)

$$y(t) = y(t_n) + \int_{t_n}^{t} f(s, y(s))\mathrm{d}s,$$

and proceed by approximating the function f with its linear interpolant on the nodes $(t_n, y(t_n))$ and $(t_{n+1}, y(t_{n+1}))$, leading to

$$f(s, y(s)) \approx f(t_n, y(t_n))\frac{s - t_{n+1}}{t_n - t_{n+1}} + f(t_{n+1}, y(t_{n+1}))\frac{s - t_n}{t_{n+1} - t_n}.$$

Hence,

$$\int_{t_n}^{t_{n+1}} f(s, y(s))\mathrm{d}s \approx \frac{h}{2}\left(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1}))\right),$$

obtaining

$$y_{n+1} = y_n + \frac{h}{2}\left(f_n + f_{n+1}\right), \tag{3.4}$$

that is the so-called *second order Adams-Moulton method*. Actually, this implicit method is not a novelty for us, since it is the trapezoidal method (2.33).

3.2 Handling Implicitness by Fixed Point Iterations

Looking at the coefficient β_k in (3.1) allows us to distinguish whether the method is explicit or implicit: indeed, if $\beta_k = 0$, the method is explicit; when $\beta_k \neq 0$, the method is implicit. As explained in Chap. 2 for one-step methods, we now aim to handle the implicitness of LMMs via fixed point iterations. To this purpose, let us first recast (3.1) in a different, equivalent form. In particular, let us separate the implicit part from the explicit one in the method, by isolating the terms for $j = k$ in the summations, leading to

$$y_{n+k} = h\beta_k f(t_{n+k}, y_{n+k}) + g_{n+k-1}, \tag{3.5}$$

where

$$g_{n+k-1} = h \sum_{j=0}^{k-1} \beta_j f_{n+j} - \sum_{j=0}^{k-1} \alpha_j y_{n+j}.$$

Then, we treat the implicitness of (3.5) by fixed point iterations as follows: in order to advance from t_n to t_{n+1},

- we arbitrarily choose an initial guess $y_{n+k}^{[0]} \in \mathbb{R}^d$. To speed up the convergence of the iterative process, a smart choice may be $y_{n+k}^{[0]} = y_{n+k-1}$;
- we perform fixed point iterations

$$y_{n+k}^{[\nu]} = h\beta_k f(t_{n+k}, y_{n+k}^{[\nu-1]}) + g_{n+k-1}, \quad \nu \geq 1, \tag{3.6}$$

stopping at the iteration M if

$$\|y_{n+k}^{[M]} - y_{n+k}^{[M-1]}\| \leq tol,$$

being tol an a-priori prescribed accuracy. Then $y_{n+k} = y_{n+k}^{[M]}$.

A major issue to address regards the convergence of the above fixed point iterative process. This aspect is object of the following result.

Theorem 3.1 *Consider the initial value problem (1.1), whose vector field satisfies the Lipschitz condition (1.8), and denote by L the Lipschitz constant. If*

$$h|\beta_k|L < 1, \tag{3.7}$$

then (3.5) has a unique solution y_{n+k} such that

$$y_{n+k} = \lim_{\nu \to \infty} y_{n+k}^{[\nu]},$$

with $y_{n+k}^{[\nu]}$ defined in (3.6), for any arbitrarily chosen initial guess $y_{n+k}^{[0]} \in \mathbb{R}^d$.

Proof Let us introduce the auxiliary map $\varphi : \mathbb{R}^d \to \mathbb{R}^d$, defined by

$$\varphi(y) = h\beta_k f(t_{n+k}, y) + g_{n+k-1}, \quad y \in \mathbb{R}^d.$$

For any $y, z \in \mathbb{R}^d$, we have

$$\|\varphi(y) - \varphi(z)\| = h\,|\beta_k|\,\|f(t_{n+k}, y) - f(t_{n+k}, z)\|$$

and, by the Lipschitz continuity of f, we obtain

$$\|\varphi(y) - \varphi(z)\| \leq h\,|\beta_k|\,L\|y - z\|.$$

Since $h|\beta_k|L < 1$,

$$\|\varphi(y) - \varphi(z)\| \leq \|y - z\|,$$

i.e., φ is a contraction in $\mathbb{R}^d$. Hence, by the contraction mapping theorem, there exists a unique fixed point $\alpha \in \mathbb{R}^d$ such that $\alpha = \varphi(\alpha)$. Since $y_{n+k} = \varphi(y_{n+k})$, y_{n+k} is the unique fixed point of the map φ. By the contraction mapping theorem, such a fixed point is the limit of the fixed point iterations

$$y_{n+k}^{[\nu]} = \varphi(y_{n+k}^{[\nu-1]}), \quad \nu \geq 1,$$

for any arbitrarily chosen initial guess $y_{n+k}^{[0]} \in \mathbb{R}^d$. □

We highlight that (3.7) is the first limitation on the stepsize we have encountered so far: in order to have a convergent fixed point iterative process for implicit LMMs (3.1), h cannot be arbitrarily chosen, but it should satisfy the restriction

$$h < \frac{1}{|\beta_k|L}.$$

We present other relevant stepsize restrictions in the remainder of this book. We will see that, in certain situations, important properties of numerical methods can be translated into proper stepsize restrictions.

3.3 Consistency and Order Conditions

Let us now focus our attention on the analysis of the accuracy of LMMs (3.1). We have seen in Chap. 2 that basic necessary accuracy and stability requirements (i.e., consistency, zero-stability and convergence) have to be fulfilled by any numerical method for (1.1). Our aim is now devoted to developing a theory of multistep methods inspired by the principles presented in the previous chapter. Hence, let us start with the following definition.

Definition 3.2 For a given grid function

$$v : \mathcal{I}_h \to \mathbb{R}^d,$$

let us denote by v_n its value in $t_n \in \mathcal{I}_h$. We define the *numerical residual operator* associated to (3.1) as

$$R_h(v_n) = \frac{1}{h}\sum_{j=0}^{k} \alpha_j v_{n+j} - \sum_{j=0}^{k} \beta_j f_{n+j}, \tag{3.8}$$

for $n = 0, 1, \ldots, N-k$, with $f_{n+j} = f(t_{n+j}, v_{n+j})$.

As it happens for one-step methods, when the residual operators (2.24) and (3.8) are respectively evaluated in the exact solution of (1.1) and its numerical approximation computed by (3.1), we have

$$R(y) = 0, \quad R_h(y_n) = 0.$$

The numerical residual operator (3.8) evaluated in the exact solution $y(t)$ of (1.1) provides the local truncation error associated to (3.1), having the following expression:

$$\begin{aligned} T(t_n, y(t_n); h) = R_h(y(t_n)) &= \frac{1}{h}\sum_{j=0}^{k} \alpha_j y(t_{n+j}) - \sum_{j=0}^{k} \beta_j f(t_{n+j}, y(t_{n+j})) \\ &= \frac{1}{h}\sum_{j=0}^{k} \alpha_j y(t_{n+j}) - \sum_{j=0}^{k} \beta_j y'(t_{n+j}). \end{aligned} \tag{3.9}$$

Correspondingly, we give the following definition.

Definition 3.3 A linear multistep method (3.1) is *consistent* if, for any $(t, y) \in [t_0, T] \times \mathbb{R}^d$,

$$\lim_{h \to 0} T(t, y; h) = 0.$$

Clearly, a consistent method (3.1) satisfies

$$T(t, y; h) = O(h^p),$$

with $p \geq 1$. In other terms, consistent methods have at least order 1. The notion of order of a LMM is given in the following definition.

Definition 3.4 A linear multistep method (3.1) has *order* p if, for a chosen vector norm $\|\cdot\|$, there exists a real constant $C > 0$ such that

$$\|T(t, y; h)\| \leq Ch^p,$$

for any $(t, y) \in [t_0, T] \times \mathbb{R}^d$, where C is independent on t, y and h.

It is the right moment to address a very crucial point. Analyzing accuracy and stability properties of numerical methods can be very tricky if we only rely on their definitions. However, the work of the pioneers of the numerical approximation of ODEs has led to highly effective tools which make the analysis much simpler, since it only requires algebraic computation involving the coefficients of the methods. As regards LMMs, seminal contributions in this direction have been provided through the talent and the ingenious work of Germund Dahlquist (1925–2005). Let us briefly present his biography, based on the obituary written by Åke Björck, Bill Gear and Gustaf Söderlind in Siam News of May 1st, 2005 and on the information reported in the gifted MacTutor History of Mathematics Archive (https://mathshistory.st-andrews.ac.uk/Biographies/Dahlquist/).

A Portrait of Germund Dahlquist

Germund Dahlquist is one of the pioneers in establishing a theory for the numerical discretization of differential equations. He was born in 1925 in Uppsala, son of a minister in the Church of Sweden (his father) and a poet (his mother). He studied mathematics at Stockholm University since 1942 and was strongly influenced by one of his professors, Harald Bohr (brother of Niels Bohr, the famous Danish physicist who achieved the Nobel Prize in 1922). Bohr was a refugee from Denmark during the Second World War and inspired Dahlquist a lot not only as regards Mathematics, but also because of his gifted character that made of him a professor highly dedicated to his students (as well as a gifted soccer player. He was member of the Danish national football team and achieved a silver medal in the 1908 Summer Olympics).

He graduated in 1949, but he did not promptly start a Ph.D. program: indeed, he was appointed at the Swedish Board of Computer Machinery as an

(continued)

applied mathematician and programmer. In 1951 Sweden developed a digital computer (its name was BESK, the acronym for Binary Electronic Sequential Calculator), which came into operation in December 1953: 1951 is a crucial year for us, since Germund Dahlquist started his studies leading to *groundbreaking contributions to the theory of numerical methods for initial value problems in ordinary differential equations*, as written in his obituary.

BESK was employed by Dahlquist to solve differential equations, clearly after a proper study of difference methods. His theoretical study was also accompanied by his membership in with a team working on numerical weather forecasts, guided by the Swedish-American meteorologist Carl Gustaf Rossby at the International Meteorological Institute of Stockholm University. This working group was able to develop in 1954 the first 24-hour weather observations made the same day, carried out on BESK.

That was a very fruitful time for Dahlquist research activity, which ted to his first publications in numerical analysis. In 1956, Dahlquist presented his studies on linear multistep methods, giving rise to the beautiful convergence theory for such methods. His theory parallels that of Peter Lax, that introduced his equivalence principle in 1955 had established the Lax principle.

From 1956 to 1959 Dahlquist covered the position head of Mathematical Analysis and Programming Development at the Swedish Board of Computer Machinery. He defended his Ph.D. thesis in 1958, entitled "Stability and Error Bounds in the Numerical Solution of Ordinary Differential Equations", advised by Fritz Carlson. In his thesis he also introduced the logarithmic norm, independently developed also by Lozinskii in 1958, explained in Definition 1.3. The theory introduced in these years was spread out by Peter Henrici in 1962, through his monograph, a masterpiece for the modern theory of numerical discretization of ODEs.

In 1959 Dahlquist he was appointed to the Royal Institute of Technology in Stockholm, where he spent the rest of his career and where the Department of Numerical Analysis and Computer Science was founded in 1962 as an offshoot the Department of Applied Mathematics. In these years he was pioneer also in establishing the new-born journal BIT Numerical Mathematics, published for the first time in 1961, served by Dahlquist as editor for more than 30 years. BIT published several relevant contributions by Germund Dahlquist, such as the highly cited masterpiece on A-stability (a concept we will later introduce in next chapters) on the famous Dahlquist barriers.

In 1963 he got his position as Full Professor of "Computer Sciences, in particular Numerical Analysis", actually the first full professorship position of this kind in Sweden. His highly appreciated book *Numeriska Metoder*, coauthored by Åke Björck appeared in 1969 and a revised extended version entitled "Numerical Methods" was published in 1974 by Prentice-Hall [113]. This book had a great success all over the world: it was translated in German

(continued)

in 1972, in Polish in 1983, in Chinese in 1990. Nick Higham writes in his review of the book:

> This work is a monumental undertaking and represents the most comprehensive textbook survey of numerical analysis to date. It will be an important reference in the field for many years to come.

During the 1960s and 1970s, Dahlquist visited many institutes in Europe, USA, Australia, New Zealand and China. He visited Stanford University in 1968 and 1977–1978, where he held a five-year part-time position from 1982 to 1986.

The Society for Industrial and Applied Mathematics (SIAM) named him their John von Neumann lecturer in 1988. Germund Dahlquist retired from the Royal Institute of Technology in 1990, but continued in actively working on research. He obtained honorary doctorates from Hamburg University (1981), Helsinki University (1994), and Linköping University (1996). In 1999 he achieved the prestigious Peter Henrici Prize, with the following motivation:

> He has created the fundamental concepts of stability, A-stability and the nonlinear G-stability for the numerical solution of ordinary differential equations. He succeeded, in an extraordinary way, to relate stability concepts to accuracy and proved the deep results which are nowadays called the first and second Dahlquist barrier. His interests, like Henrici's, are very broad, and he contributed significantly to many parts of numerical analysis. As a human being and scientist, he gives freely of his talent and knowledge to others and remains a model for many generations of scientists to come.

In 1995, on the occasion of his 70th birthday, SIAM established the Germund Dahlquist Prize to be awarded biennially, normally to a young scientist for original contributions to fields related to the numerical solution of differential equations and numerical methods for scientific computing.

In his obituary, two more significant aspects arises: his active work for Amnesty International and his love of music. Here is an excerpt:

> As an active member of Amnesty International during the 1970s, Dahlquist worked to help scientists who were politically persecuted, in some cases traveling to offer his encouragement and recognition in person. He used to tell the story of his intervention on behalf of a Russian mathematician who, in despair, had made a thoughtless public statement to the effect that the Soviet Union was "a land of alcoholics". Guriy I Marchuk, who had visited Stockholm University in the 1960s, was then president of the USSR Academy of Sciences and vice-chair of the USSR Council of Ministers. Dahlquist wrote to Marchuk pleading the dissident's case. After a long time with no response, two staff members of the Soviet Embassy called at Germund's office one day, bringing greetings from Marchuk and a package, that turned out to contain... two bottles of vodka! Germund had a keen interest in music, mainly classical but also jazz music. He would often happily sit down at the piano and entertain his colleagues with a few old standards, starting with "On the Sunny Side of the Street" and ending with "As Time Goes By". But his knowledge went much deeper. On one visit to the

(continued)

USA, with a few colleagues in a fine restaurant, Germund heard a female bar pianist whose music was obviously the highlight of the evening for him. When it was time to leave, Germund told the pianist how much he had enjoyed her stylish playing, adding that it had reminded him of one of his favorites, the great jazz pianist Art Tatum. The pianist was duly flattered, but it was Germund who was surprised when she answered: "Art Tatum was my father!".

Germund Dahlquist died in 2005. His work inspired many researches over several decades. As John Butcher wrote:

When I met him in 1970, I started to appreciate that he was more than a brilliant mathematician and computational scientist: he was a kind and sensitive man and a loyal friend. [...] Everything Germund published was a separate gem, exhibiting deep mathematical insight and, at the same time, a clear understanding of sound computational practice. He was a pioneer who remained a central figure throughout his career; he will be sadly missed.

Let us recast (3.9) in the following form

$$\sum_{j=0}^{k} \alpha_j y(t_{n+j}) - h \sum_{j=0}^{k} \beta_j y'(t_{n+j}) = hT(t_n, y(t_n); h).$$

Such a relation, though defined on vector valued functions, actually results to be the same on each component of the involved vectors. For this reasons, it makes sense to analyze it on scalar functions, motivating the following definition.

Definition 3.5 For a given scalar function $z : [t_0, T] \to \mathbb{R}$ of class $C^1([t_0, T])$, the *linear difference operator* associated to a linear multistep method (3.1) is defined by

$$\mathcal{L}[z(t), h] = \sum_{j=0}^{k} \alpha_j z(t + jh) - h \sum_{j=0}^{k} \beta_j z'(t + jh). \tag{3.10}$$

Clearly, if $y(t)$ is the solution of (1.1), we have

$$\mathcal{L}[y_i(t), h] = hT(t, y_i(t); h), \quad i = 1, 2, \ldots, d$$

and, as a consequence, if the method (3.1) has order p, $\mathcal{L}[z(t), h] = O(h^{p+1})$, for any scalar function $z(t)$. This observation leads to the following result.

Theorem 3.2 *A linear multistep method* (3.1) *has order* p *if and only if* $C_\ell = 0$, *for any* $\ell = 0, 1, \ldots, p$, *with*

$$C_0 = \sum_{j=0}^{k} \alpha_j, \qquad C_\ell = \sum_{j=0}^{k} \left(\frac{j^\ell}{\ell!} \alpha_j - \frac{j^{\ell-1}}{(\ell-1)!} \beta_j \right), \ \ell > 0$$

and $C_{p+1} \neq 0$.

Proof We expand in Taylor series around t each evaluation of $z(t)$ appearing in the right-hand side of (3.10), obtaining

$$\mathcal{L}[z(t), h] = \sum_{j=0}^{k} \alpha_j \left(z(t) + \sum_{\ell \geq 1} \frac{(jh)^\ell}{\ell!} z^{(\ell)}(t) \right) - h \sum_{j=0}^{k} \beta_j \left(z'(t) + \sum_{\ell > 1} \frac{(jh)^{\ell-1}}{(\ell-1)!} z^{(\ell)}(t) \right).$$

Collecting in powers of h leads to

$$\mathcal{L}[z(t), h] = \left(\sum_{j=0}^{k} \alpha_j \right) z(t) + \sum_{\ell \geq 1} \sum_{j=0}^{k} \left(\frac{j^\ell}{\ell!} \alpha_j - \frac{j^{\ell-1}}{(\ell-1)!} \beta_j \right) h^\ell z^{(\ell)}(t),$$

i.e.,

$$\mathcal{L}[z(t), h] = C_0 z(t) + \sum_{\ell \geq 1} C_\ell h^\ell z^{(\ell)}(t).$$

In order to have $\mathcal{L}[z(t), h] = O(h^{p+1})$, we need to satisfy

$$C_0 = C_1 = \cdots = C_p = 0, \quad C_{p+1} \neq 0,$$

obtaining the thesis. □

Definition 3.6 The non-zero constant

$$C_{p+1} = \sum_{j=0}^{k} \left(\frac{j^{p+1}}{(p+1)!} \alpha_j - \frac{j^p}{p!} \beta_j \right)$$

of an order p method (3.1) is denoted as its *error constant*.

In other terms, for an order p linear multistep method we have

$$\mathcal{L}[z(t), h] = C_{p+1} h^{p+1} z^{(p+1)}(t) + O(h^{p+2}),$$

for all regular scalar functions $z(t)$, while the corresponding local truncation error is

$$T(t, y(t); h) = C_{p+1} h^p y^{(p+1)}(t) + O(h^{p+1}).$$

The following corollary of Theorem 3.2 gives us a very immediate way to analyze the consistency of a LMM (3.1), only requiring a straightforward algebraic computation involving the coefficients of the method. This way of proceeding is certainly much simpler than proving consistency using Definition 3.3.

Corollary 3.1 *A linear multistep method* (3.1) *is consistent if and only if*

$$\sum_{j=0}^{k} \alpha_j = 0, \quad \sum_{j=0}^{k} (j\alpha_j - \beta_j) = 0. \tag{3.11}$$

Proof A consistent method has order at least one. Hence, according to Theorem 3.2, it satisfies

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \quad C_1 = \sum_{j=0}^{k} (j\alpha_j - \beta_j) = 0$$

and the thesis holds true. □

Let us now present few examples of applications of the notions introduced in this section.

Example 3.4 Let us analyze consistency, orders and error constants of the examples of LMMs (3.1) we have developed so far, depending on one and two steps.

- The explicit Euler method (2.19) satisfies

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \quad C_1 = \sum_{j=0}^{k} \left(j\alpha_j - \beta_j\right) = 0,$$

$$C_2 = \sum_{j=0}^{k} \left(\frac{j^2}{2}\alpha_j - j\beta_j\right) = \frac{1}{2},$$

  so it is consistent, of order 1 and its error constant is equal to $1/2$;
- for the implicit Explicit Euler method (2.32) we have

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \quad C_1 = \sum_{j=0}^{k} \left(j\alpha_j - \beta_j\right) = 0,$$

$$C_2 = \sum_{j=0}^{k} \left(\frac{j^2}{2}\alpha_j - j\beta_j\right) = -\frac{1}{2}.$$

  Then, it is consistent, of order 1 and its error constant is equal to $-1/2$;
- the trapezoidal method (2.33) fulfills

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \qquad C_1 = \sum_{j=0}^{k} \left(j\alpha_j - \beta_j\right) = 0,$$

$$C_2 = \sum_{j=0}^{k} \left(\frac{j^2}{2}\alpha_j - j\beta_j\right) = 0, \quad C_3 = \sum_{j=0}^{k} \left(\frac{j^3}{6}\alpha_j - \frac{j^2}{2}\beta_j\right) = -\frac{1}{12},$$

  so it is consistent, of order 2 and its error constant is $-1/12$;
- Milne-Simpson method (3.2) is a LMM (3.1) with $\alpha_0 = -1$, $\alpha_1 = 0$,

(continued)

Example 3.4 (continued)
$\alpha_2 = 1$, $\beta_0 = 1/3$, $\beta_1 = 4/3$ and $\beta_2 = 1/3$. Hence, it satisfies

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \qquad C_1 = \sum_{j=0}^{k} (j\alpha_j - \beta_j) = 0,$$
$$C_2 = \sum_{j=0}^{k} \left(\frac{j^2}{2}\alpha_j - j\beta_j \right) = 0, \qquad C_3 = \sum_{j=0}^{k} \left(\frac{j^3}{6}\alpha_j - \frac{j^2}{2}\beta_j \right) = 0,$$
$$C_4 = \sum_{j=0}^{k} \left(\frac{j^4}{24}\alpha_j - \frac{j^3}{6}\beta_j \right) = 0, \qquad C_5 = \sum_{j=0}^{k} \left(\frac{j^5}{120}\alpha_j - \frac{j^4}{24}\beta_j \right) = -\frac{1}{90},$$

so it is consistent, of order 4, with error constant $-1/90$;

- the two-step Adams-Bashforth method (3.3) is a LMM (3.1) with $\alpha_0 = 0$, $\alpha_1 = -1$, $\alpha_2 = 1$, $\beta_0 = -1/2$, $\beta_1 = 3/2$ and $\beta_2 = 0$. Therefore,

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \qquad C_1 = \sum_{j=0}^{k} (j\alpha_j - \beta_j) = 0,$$
$$C_2 = \sum_{j=0}^{k} \left(\frac{j^2}{2}\alpha_j - j\beta_j \right) = 0, \qquad C_3 = \sum_{j=0}^{k} \left(\frac{j^3}{6}\alpha_j - \frac{j^2}{2}\beta_j \right) = \frac{5}{12},$$

so it is consistent, of order 2 and its error constant is equal to $5/12$.

Example 3.5 We now aim to study the consistency of the following numerical method

$$y_{n+2} - 2y_{n+1} + y_n = hf_n, \tag{3.12}$$

both using conditions (3.11) and through an experimental check. Equation (3.12) provides an explicit two-step method and, according to Corollary 3.1 it is a non-consistent, since

$$C_0 = \sum_{j=0}^{k} \alpha_j = 0, \qquad C_1 = \sum_{j=0}^{k} (j\alpha_j - \beta_j) = -1.$$

(continued)

Example 3.5 (continued)
Let us experimentally check this property. To this purpose, consider the following scalar test problem

$$\begin{cases} y'(t) = 2(y(t) - \cos(t)) - \sin(t), & t \in [0, 2\pi], \\ y(0) = 1, \end{cases} \tag{3.13}$$

whose exact solution is $y(t) = \cos(t)$. As visible in Fig. 3.1, the numerical solution does not match the exact one and the reader can verify that a similar behavior occurs also if the stepsize is reduced. This is a typical situation of lack of consistency: the application of a non-consistent method leads to the pattern of another function rather than one reproducing the exact solution.

3.4 Zero-Stability

We have learned in Chap. 2 that consistency is a local accuracy property, while zero-stability and convergence are global accuracy properties. We now focus on zero-stability analysis that, according to our analysis in Chap. 2, ensures the boundedness of the error for small values of the stepsize. We first need to introduce the following tools.

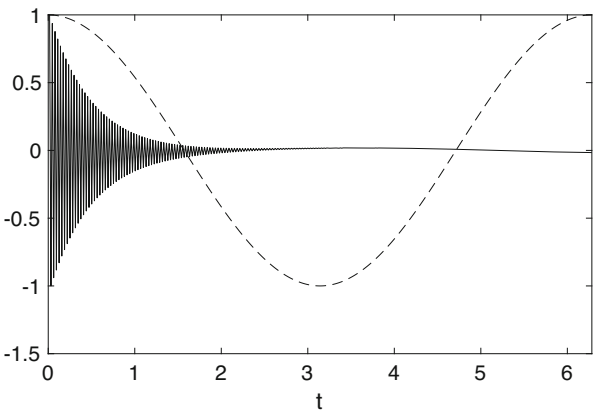


Fig. 3.1 Numerical (straight line) vs exact (dashed line) solutions of (3.13). The numerical solution is computed by the non-consistent method (3.12) with stepsize $h = \pi/100$

Definition 3.7 The *first characteristic polynomial* associated to a linear multistep method (3.1) is given by

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j, \tag{3.14}$$

while the *second characteristic polynomial* associated to (3.1) is given by

$$\sigma(z) = \sum_{j=0}^{k} \beta_j z^j. \tag{3.15}$$

These polynomials are very useful also to analyze consistency of (3.1). Indeed, for a consistent method, we have

$$\rho(1) = 0, \quad \rho'(1) = \sigma(1),$$

since

$$\rho(1) = \sum_{j=0}^{k} \alpha_j, \quad \rho'(1) = \sum_{j=0}^{k} j\alpha_j, \quad \sigma(1) = \sum_{j=0}^{k} \beta_j.$$

The roots of the first characteristic polynomial are important to analyze the zero-stability of (3.1). To this purpose, there is a relevant property that we are going to use, defined as follows.

Definition 3.8 An algebraic polynomial satisfies the *root condition* if each of its roots has modulus strictly less than 1 or has modulus one but it is simple.

Example 3.6 The polynomial

$$\rho(z) = z^2 - \frac{5}{6}z + \frac{1}{6}$$

satisfies the root condition, since its roots are 1/2 and 1/3. The polynomial

(continued)

Example 3.6 (continued)

$$\rho(z) = z^2 - \frac{4}{3}z + \frac{1}{3}$$

also satisfies the root condition, since its roots are 1/3 and 1. Finally, the polynomial

$$\rho(z) = z^3 - \frac{5}{2}z^2 + 2z - \frac{1}{2}$$

does not satisfy the root condition, since its roots are 1/2 and 1, the latter with multiplicity 2.

As we know from Chap. 2, zero-stability ensures that the numerical solution does not blow-up as h tends to 0; in other terms, we need to prove that the general solution of the difference equation describing (3.1) does not blow-up as h goes to 0. Hence, we first have to analyze what happens to the solution of a linear difference equation; this issue is explained by the following result, reported for the scalar case, whose proof is here omitted (the interested reader can refer, for instance, to [170]).

Theorem 3.3 *Consider the following order k inhomogeneous linear difference equation*

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = g_{n+k},$$

where $\alpha_j, y_{n+j} \in \mathbb{R}$, $j = 0, 1, \ldots, k$, *and* $g_{n+k} \in \mathbb{R}$. *Then, there exists* $M > 0$ *independent on n such that*

$$|y_n| \leq M \left(\max_{0 \leq i \leq k-1} |y_i| + \sum_{m=k}^{n} |g_m| \right), \quad n \geq 0,$$

if and only if the characteristic polynomial

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j$$

satisfies the root condition.

According to Theorem 3.3, the root condition is a necessary and sufficient condition for the stability of the solutions of linear difference equations. We now prove that the root condition of the first characteristic polynomial (3.14) is exactly what we need to have stable numerical solutions through LMMs (3.1). First of all, let us provide a rigorous definition of zero-stability.

Consider the vector

$$\mathbf{y}_h = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_N \end{bmatrix} \tag{3.16}$$

collecting the numerical solution of the initial value problem (1.1) in each grid point, computed by (3.1) and the vector

$$\widetilde{\mathbf{y}}_h = \begin{bmatrix} \widetilde{y}_0 \\ \widetilde{y}_1 \\ \vdots \\ \widetilde{y}_N \end{bmatrix}$$

of the numerical approximations of the solution to the perturbed problem (1.12), obtained by (3.1). As in the one-step case described in Chap. 2, $R_h(y_n) = 0$, while we suppose that $R_h(\widetilde{y}_n) = \varepsilon_n$ and collect all these values in the vector

$$\varepsilon = \begin{bmatrix} \varepsilon_0 \\ \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{bmatrix}.$$

We also denote by δ the vector collecting the deviations in the initial values, i.e.,

$$\delta = \begin{bmatrix} y_0 - \widetilde{y}_0 \\ y_1 - \widetilde{y}_1 \\ \vdots \\ y_{k-1} - \widetilde{y}_{k-1} \end{bmatrix}.$$

We define zero-stability of LMMs (3.1) as follows.

Definition 3.9 A linear multistep method (3.1) is *zero-stable* if there exists $\Lambda > 0$ such that, for any value of the stepsize $h \in [0, h_0]$, with $h_0 > 0$, the following stability inequality holds true

$$\|\mathbf{y}_h - \widetilde{\mathbf{y}}_h\|_\infty \leq \Lambda(\|\delta\|_\infty + \|\varepsilon\|_\infty). \tag{3.17}$$

As already observed in Chap. 2 for one-step methods, the zero-stability inequality (3.17) imitates the corresponding inequality (1.15), useful to study the continuous dependence on the initial data and the vector field of the underlying initial value problem.

Let us now prove the following zero-stability criterion, whose statement recalls the stability result for difference equations presented in Theorem (3.3).

Theorem 3.4 *A linear multistep method* (3.1) *is zero-stable if and only if its first characteristic polynomial* (3.14) *satisfies the root condition.*

Proof We separately prove the necessity and the sufficiency of the root condition for the zero-stability of (3.1).

- First part: let us prove that zero-stability implies the root condition for its first characteristic polynomial. Suppose that the vector field f of the continuous problem (1.1) is identically null. The corresponding LMM (3.1) reads

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = 0. \tag{3.18}$$

  Let us collect the numerical solution arising from (3.18) in the vector $\mathbf{y}_h$ given by (3.16) and consider that the homogeneous difference equation (3.18) also admits the zero solution. Then, from the zero-stability hypothesis, there exists $\Lambda > 0$ such that

$$\|\mathbf{y}_h\|_\infty \leq \Lambda \max_{0 \leq j \leq k-1} |y_j|.$$

  Hence, the solution $\mathbf{y}_h$ of the difference equation (3.18) is uniformly bounded and, by Theorem 3.3, its characteristic polynomial

$$\rho(z) = \sum_{j=0}^{k} \alpha_j z^j$$

satisfies the root condition.

- Second part: let us prove that the root condition for the first characteristic polynomial of (3.1) implies its zero-stability. Let us consider two given grid functions

$$u, v : \mathcal{I}_h \to \mathbb{R}^d$$

and denote by u_n and v_n their values in $t_n \in \mathcal{I}_h$, respectively. Correspondingly,

$$\sum_{j=0}^{k} \alpha_j u_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, u_{n+j}) + h R_h(u_n),$$
$$\sum_{j=0}^{k} \alpha_j v_{n+j} = h \sum_{j=0}^{k} \beta_j f(t_{n+j}, v_{n+j}) + h R_h(v_n).$$

By subtraction, we have

$$\sum_{j=0}^{k} \alpha_j \left(u_{n+j} - v_{n+j}\right) = g_{n+k}, \tag{3.19}$$

where

$$g_{n+k} = h \sum_{j=0}^{k} \beta_j \left(f(t_{n+j}, u_{n+j}) - f(t_{n+j}, v_{n+j})\right) + h \left(R_h(u_n) - R_h(v_n)\right).$$

In other terms, $\{u_n - v_n\}_{n\in\mathbb{N}}$ is solution of the inhomogeneous difference equation (3.19). Since the root condition holds true, by Theorem 3.3 there exists $M > 0$, independent on n, such that

$$\|u_n - v_n\|_\infty \le M \left(\max_{0\le i\le k-1} \|u_i - v_i\|_\infty + \sum_{m=k}^{n} \|g_m\|_\infty \right).$$

Certainly, by defining $(r_h)_n = R_h(u_n) - R_h(v_n)$, we have

$$\|g_m\|_\infty = \left\| h \sum_{j=0}^{k} \beta_j \left(f(t_{m+j-k}, u_{m+j-k}) - f(t_{m+j-k}, v_{m+j-k})\right) + h(r_h)_{m-k} \right\|_\infty$$
$$\le hL\beta \sum_{j=0}^{k} \|u_{m+j-k} - v_{m+j-k}\|_\infty + h\|r_h\|_\infty,$$

being $\beta = \max_{0 \le j \le k} \beta_j$. By defining $e_n = u_n - v_n$, we have

$$\|e_n\|_\infty \le M \left(\max_{0 \le i \le k-1} \|e_i\|_\infty + \sum_{m=k}^{n} \left(h\ L\ \beta \sum_{j=0}^{k} \|e_{m+j-k}\|_\infty + h\|r_h\|_\infty \right) \right)$$

$$\le M \left(\max_{0 \le i \le k-1} \|e_i\|_\infty + h\ L\ \beta \sum_{m=k}^{n} \sum_{j=0}^{k} \|e_{m+j-k}\|_\infty + Nh\|r_h\|_\infty \right).$$

Clearly,

$$\sum_{m=k}^{n} \sum_{j=0}^{k} \|e_{m+j-k}\|_\infty \le \sum_{j=0}^{k} \sum_{m=0}^{n} \|e_m\|_\infty = (k+1) \sum_{m=0}^{n} \|e_m\|_\infty.$$

Hence,

$$\|e_n\|_\infty \le M \left(\max_{0 \le i \le k-1} \|e_i\|_\infty + h\ L\ \beta(k+1) \sum_{m=0}^{n} \|e_m\|_\infty + (T - t_0)\|r_h\|_\infty \right)$$

or, equivalently,

$$\mu\|e_n\|_\infty \le M \left(\max_{0 \le i \le k-1} \|e_i\|_\infty + h\ L\ \beta(k+1) \sum_{m=0}^{n-1} \|e_m\|_\infty + (T - t_0)\|r_h\|_\infty \right),$$

with $\mu = (1 - h\ L\ \beta(k+1))$. Suppose to choose h small enough in order to make $\mu > 0$ (it is enough to choose $h < 1/(hL\beta(k+1))$ to make this possible), so that

$$\|e_n\|_\infty \le h\ A \sum_{m=0}^{n-1} \|e_m\|_\infty + B,$$

with

$$A = \frac{M}{\mu} L\ \beta(k+1), \quad B = \frac{M}{\mu} \left(\max_{0 \le i \le k-1} \|e_i\|_\infty + (T - t_0)\|r_h\|_\infty \right).$$

Let us consider the corresponding difference equation

$$w_n = h\ A \sum_{m=0}^{n-1} w_m + B,$$

with initial value $w_0 = B$. The reader can easily prove by induction that its solution is

$$w_n = B(1 + hA)^n, \quad n \geq 0.$$

Then

$$\|e_n\|_\infty - w_n \leq h\, A \sum_{m=0}^{n-1} (\|e_m\|_\infty - w_m)$$

and the reader can again prove by induction that

$$\|e_n\|_\infty \leq w_n.$$

Therefore,

$$\|e_n\|_\infty \leq B(1 + hA)^n \leq B\mathrm{e}^{nhA} \leq B\mathrm{e}^{NhA} = B\mathrm{e}^{(T-t_0)A}$$

and replacing the value of B gives

$$\|e_n\|_\infty \leq \frac{M}{\mu}\mathrm{e}^{(T-t_0)A}\left(\max_{0\leq i\leq k-1}\|e_i\|_\infty + (T - t_0)\|r_h\|_\infty\right).$$

The choice

$$\Lambda = \frac{M}{\mu}\mathrm{e}^{(T-t_0)A}\max\{1, T - t_0\}$$

let the thesis hold true. □

Analyzing zero-stability through Definition 3.9 may be very tricky. However, Theorem 3.4 provides a practical condition that makes the analysis of zero-stability much easier. Let us test such a condition on few examples.

Example 3.7 According to Theorem 3.4,

- Euler methods (2.19) and (2.32) are zero-stable, since their first characteristic polynomial is

$$\rho(z) = z - 1;$$

(continued)

Example 3.7 (continued)

- the trapezoidal method (2.33) is zero-stable, since its first characteristic polynomial is

$$\rho(z) = z - 1;$$

- Milne-Simpson method (3.2) is zero-stable, since its first characteristic polynomial is

$$\rho(z) = z^2 - 1;$$

- the two-step Adams-Bashforth method (3.3) is zero-stable, since its first characteristic polynomial is

$$\rho(z) = z^2 - z.$$

Example 3.8 We now aim to study the zero-stability of the following numerical method

$$y_{n+2} - 3y_{n+1} + 2y_n = hf_n, \tag{3.20}$$

both checking the root condition and experimentally. This method is not zero-stable, since the roots of its first characteristic polynomial

$$\rho(z) = z^2 - 3z + 2$$

are 1 and 2, so the root condition is not satisfied. Let us experimentally check this lack of zero-stability on the test problem (3.13). As visible in Fig. 3.2, the numerical solution does not have a stable behavior, so it does not match the stable character of the exact solution. Hence, the applied method is clearly not zero-stable.

For a zero-stable linear multistep method (3.1), the following result holds true.

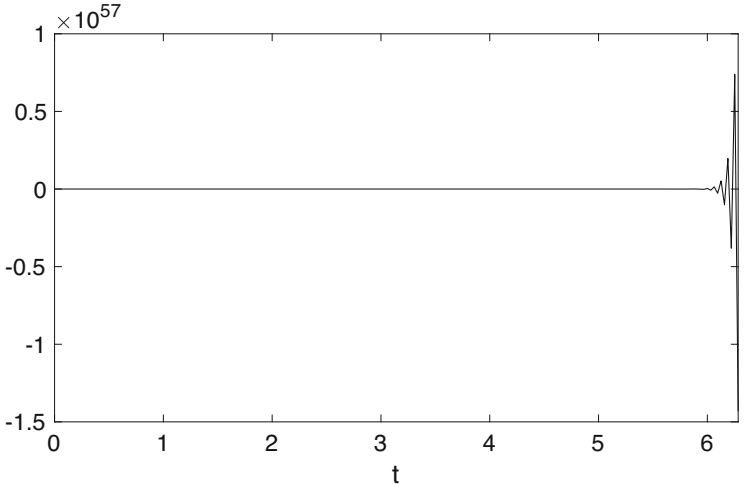


Fig. 3.2 Numerical solution of (3.13) computed by method (3.20), that is not zero-stable, with stepsize $h = \pi/100$

Theorem 3.5 *The order p of a zero-stable linear multistep method* (3.1) *depending on k steps satisfies*

$$p \leq \begin{cases} k+1, & k \text{ odd}, \\ k+2, & k \text{ even}. \end{cases}$$

This results, whose proof can be found, for instance, in [67], is an order barrier for linear multistep methods, well-known in the literature as *first Dahlquist barrier*. In other terms, the number of steps provides an upper bound for the order of convergence of the corresponding method. Maximal order methods are those of order $k+1$ if k is odd, $k+2$ if k is even. An example of maximal order method is the Milne-Simpson method (3.2), which depends on 2 steps and has order 4. However, maximal order methods can have poor stability properties, as it is the case of Milne-Simpson method itself, so we should provide a reasonable balance between accuracy and stability properties. We will analyze this aspect in Chap. 6.

3.5 Convergence

Let us now turn our attention to analysis of convergence for linear multistep methods (3.1), starting from the following definition.

Definition 3.10 Suppose that (3.16) is the vector collecting the numerical approximations of the solution to the continuous problem (1.1) in each grid point of the uniform grid (2.1), obtained by the method (3.1) and also consider the vector of the corresponding exact values

$$\boldsymbol{v}_h = \begin{bmatrix} y(t_0) \\ y(t_1) \\ \vdots \\ y(t_N) \end{bmatrix}.$$

Denote by

$$s_h = \begin{bmatrix} y_1 \\ \vdots \\ y_{k-1} \end{bmatrix}$$

the vector collecting the missing starting values computed by a proper starting procedure. Then, the LMM (3.1) is *convergent* if

$$\lim_{h \to 0} \|\boldsymbol{v}_h - \mathbf{y}_h\|_\infty = 0,$$

whenever

$$\lim_{h \to 0} (s_h)_i = y(t_i), \quad i = 1, 2, \ldots, k-1.$$

It is worth underlining that a significant role is played by the starting procedure, which is assumed to provide accurate starting values in the above given definition of convergence. Again, as aforementioned for consistency and zero-stability, convergence analysis through its definition may not be an easy task. However, there is a very powerful result, according to which convergence is equivalent to consistency plus zero-stability. In such a way, convergence analysis only relies on very simple calculations involving the coefficients of the method. This result originally belongs to a huge masterpiece due to Peter D. Lax (Budapest, 1926), specialized to LMMs by Germund Dahlquist. Peter Lax is a mathematician born in Hungary, Professor at

Courant Institute of Mathematical Sciences at New York University, winner of the prestigious Abel Prize in 2005.

Theorem 3.6 (Lax Equivalence Theorem) *A linear multistep method* (3.1) *is convergent if and only if it is consistent and zero-stable.*

Proof We separately prove the necessity and the sufficiency parts of the theorem.

- First part: we prove that convergence implies consistency and zero-stability. Consider the initial value problem (1.1) with f identically zero and $y_0 = 0$. Then, $y(t) = 0$. By contradiction, suppose that the method is not zero-stable: then, there exists are a root $\overline{z}$ of the first characteristic polynomial of (3.1) such that $|\overline{z}| > 1$ or $|\overline{z}| = 1$ and its multiplicity is greater than 1. In the first case, the solution of (3.1), thought as a difference equation, contains a term

$$c\overline{z}^n, \quad c \in \mathbb{R}$$

tending to infinity, which contradicts the hypothesis of convergence. The case $|\overline{z}| = 1$ with multiplicity greater than 1 is similar and left to the reader.
Let us now prove that convergence implies consistency, by proving that consistency conditions arise from the exactness of the method on the monomial basis $\{1, t\}$ (this choice is connected to Exercise 4, Sect. 3.6).

(*i*) Consider the initial value problem (1.1) with f identically zero and $y_0 = 1$. Then, $y(t) = 1$. The corresponding LMM (3.1) is given by

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = 0$$

and, assuming $y_1 = y_2 = \cdots = y_{k-1}$, convergence yields

$$\sum_{j=0}^{k} \alpha_j = 0,$$

that is the first consistency condition.

(*ii*) Consider the initial value problem (1.1) with f identically equal to 1 and $y_0 = 0$. Then, $y(t) = t - t_0$. The corresponding LMM (3.1) is given by

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j,$$

i.e.,

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\sigma(1). \tag{3.21}$$

Certainly,

$$y_n = \frac{\sigma(1)}{\rho'(1)} nh$$

is solution of (3.21), as the reader can easily check. By convergence, y_n converges to nh and, as a consequence,

$$\frac{\sigma(1)}{\rho'(1)} = 1$$

that is the second consistency condition.

- Second part: we prove that consistency and zero-stability imply convergence. Since $R_h(y_n) = 0$ and $R_h(y(t)) = T(t, y(t); h)$, the zero-stability inequality (3.17) reads

$$\|\boldsymbol{v}_h - \mathbf{y}_h\|_\infty \leq \Lambda(\|\delta\|_\infty + \max_{0 \leq i \leq N} |T(t_i, y(t_i); h)|).$$

The right-hand side of last inequality goes to 0, because of the preliminary assumption on the starting procedure given in Definition 3.10 and the consistency assumption, leading to the thesis. □

Example 3.9 Let us analyze the family of two-step implicit LMMs (3.1), given by

$$y_{n+2} + \alpha_1 y_{n+1} + \alpha_0 y_n = h(\beta_2 f_{n+2} + \beta_1 f_{n+1} + \beta_0 f_n). \tag{3.22}$$

- *Consistency*. Let us give consistency conditions (3.11) for this class of methods. We have

$$\alpha_0 + \alpha_1 + 1 = 0,$$

$$\alpha_1 + 2 = \beta_0 + \beta_1 + \beta_2.$$

(continued)

Example 3.9 (continued)

- *Order 2*. We obtain the constraints on the coefficients of the methods ensuring second order, by imposing the additional condition $C_2 = 0$ in Theorem (3.2), i.e.,

$$\frac{1}{2}\alpha_1 + 2 = \beta_1 + 2\beta_2.$$

Hence, order 2 methods (3.22) satisfy

$$\alpha_0 = 3 - 2\beta_1 - 4\beta_2,$$
$$\alpha_1 = -4 + 2\beta_1 + 4\beta_2,$$
$$\beta_0 = -2 + \beta_1 - 3\beta_2.$$

- *Order 3*. Third order methods also satisfy $C_3 = 0$ in Theorem (3.2), i.e.,

$$\frac{1}{6}\alpha_1 + \frac{4}{3} = \frac{1}{2}\beta_1 + 2\beta_2.$$

In summary, third order methods satisfy

$$\alpha_0 = -5 + 12\beta_2,$$
$$\alpha_1 = 4 - 12\beta_2,$$
$$\beta_0 = 2 - 11\beta_2.$$
$$\beta_1 = 4 - 8\beta_2.$$

- *Order 4*. Fourth order methods also fulfill $C_4 = 0$ in Theorem (3.2), i.e.,

$$\frac{1}{24}\alpha_1 + \frac{2}{3} = \frac{1}{6}\beta_1 + \frac{4}{3}\beta_2.$$

In summary, fourth order methods satisfy

$$\alpha_0 = 7, \quad \alpha_1 = -8, \quad \beta_0 = -9, \quad \beta_1 = -4, \quad \beta_2 = -1.$$

We observe that the maximal order method

$$y_{n+2} - 8y_{n+1} + 7y_n = -h(f_{n+2} + 4f_{n+1} + 9f_n)$$

(continued)

Example 3.9 (continued)
is not convergent because it is not-zero stable, since the zeros of the first characteristic polynomial

$$\rho(z) = z^2 - 8z + 7$$

are 1 and 7, then the root condition is not satisfied.

Clearly, due to Lax equivalence theorem, if a method is not convergent then consistency and/or zero-stability are missing. A numerical evidence of this aspect has already been given in Examples 3.5 and 3.8. We now aim to experimentally analyze the performances of a convergent method, namely the second-order Adams-Bashforth method (3.3), implemented in Program 3.1 by using the explicit Euler method (2.19) as a starting procedure.

Program 3.1 (Adams-Bashforth Method)

```
% Function implementing the second order Adams-Bashforth
% method on a uniform grid, for the numerical solution
% of a d-dimensional ODE.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%           of the integration interval;
% - y0: initial value;
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: d×N matrix whose i-th column y(:,i) stores the
%      approximate value in the i-th grid point, i=1,2,...,N.

function [t,y]=AdamsBashforth(problem,tspan,y0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0);
y=zeros(d,N);
fold=f(problem,tspan(1),y0);
% Starting value y(:,1) recovered by explicit Euler method
y(:,1)=y0+h*fold;
fnew=f(problem,t(1),y(:,1));
y(:,2)=y(:,1)-h*(fold-3*fnew)/2;
% The variables fold and fnew are introduced to reuse
% already computed function evaluations.
```

(continued)

Program 3.1 (continued)

```
fold=fnew;
fnew=f(problem,t(2),y(:,2));
for i=3:N
    y(:,i)=y(:,i-1)-h*(fold-3*fnew)/2;
    fold=fnew;
    fnew=f(problem,t(i),y(:,i));
end
```

Next example gives an experimental confirmation of the order of a numerical method. The involved order estimate is based on the application of the method with stepsize h, whose principal error term is given by $err(h) \approx C_{p+1}h^p$, and halved stepsize $h/2$, with principal error term $err(h/2) \approx C_{p+1}(h/2)^p$. The ratio between the two errors gives

$$\frac{err(h)}{err(h/2)} \approx 2^p,$$

i.e.,

$$p \approx \log_2 \left(\frac{err(h)}{err(h/2)} \right), \tag{3.23}$$

that provides an estimate of the order of convergence. Clearly, the smaller is h, the more the estimate of p is accurate.

Example 3.10 Consider the following van der Pol oscillator

$$\begin{cases} y_1'(t) = y_2(t), \\ y_2'(t) = (1 - y_1(t)^2)y_2(t) - y_1(t), \end{cases} \tag{3.24}$$

with $t \in [0, 10]$ and initial value $y_0 = [\,2 \;\; -2/3\,]^{\mathsf{T}}$. Let us numerically solve this problem by using the second order Adams-Bashforth method (3.3): the pattern of the solution is displayed in Fig. 3.3. We know that (3.4) is a convergent method; let us give an experimental evidence of this property in Table 3.1, where it is visible that the more the stepsize diminishes, the more the error decreases. The error is computed as

$$\|y_{\text{AB}} - y_{\text{ODE45}}\|_\infty,$$

(continued)

Example 3.10 (continued)
where $y_{AB} \approx y(10)$ is computed by (3.3) and y_{ODE45} is the solution in $t = 10$ computed by the Matlab built-in function `ode45`, with high accuracy, given by

$$y_{ODE45} = \left[-1.914027891764918 \quad 0.446099168376746 \right]^{\mathsf{T}}.$$

As visible, both the convergence and the order of convergence are confirmed in the experiments.

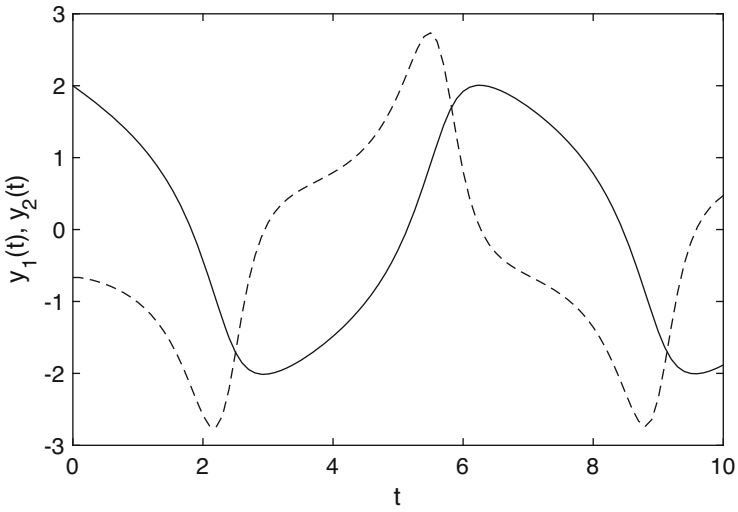


Fig. 3.3 Pattern of the solution of (3.24) computed by the Adams-Bashforth method (3.3), with $h = 0.1$. The plot of the component $y_1(t)$ is the straight line, while $y_2(t)$ is the dashed line

Table 3.1 Example 3.10: error in the final integration point associated to the application of the Adams-Bashforth method (3.4) to the van der Pol problem (3.24) and order estimation, computed as suggested by Eq. (3.23)

h	$\|y_{AB} - y_{ODE45}\|_\infty$	p
$h_0 = 0.1$	$3.10 \cdot 10^{-2}$	
$h_0/2$	$8.23 \cdot 10^{-3}$	1.91
$h_0/4$	$2.13 \cdot 10^{-3}$	1.95
$h_0/8$	$5.41 \cdot 10^{-4}$	1.97
$h_0/16$	$1.36 \cdot 10^{-4}$	1.99

3.6 Exercises

1. Analyze the family of explicit three-step methods (3.1), providing families of convergent methods of various orders. Is the maximal order method of the family also convergent?
2. Compute the values of $\vartheta_1, \vartheta_2 \in \mathbb{R}$ such that the two-parameter family of methods

$$y_{n+2} + (2\vartheta_1 - 3\vartheta_2)y_{n+1} - \left(\frac{5\vartheta_1}{2} - 2\right) y_n = \vartheta_1 h\,(f_n + f_{n+1})$$

   reduces to a single convergent method. Does the corresponding method also achieve maximal order?
3. As seen in Example 3.4, Milne-Simpson method (3.2) is an implicit two-step method of order 4. Provide an experimental confirmation of its order on the van der Pol problem (3.24).
4. Prove that a LMM (3.1) of order p exactly solves all the differential problems whose solution is a polynomial of degree at most p.
   Hint: prove that, for methods of order p, the linear difference operator (3.10) annihilates on the set of functions $\{1,\ t,\ t^2,\ \ldots,\ t^p\}$.
5. Construct an explicit two-step method (3.1) exactly solving all differential problems whose solution belongs to the functional space spanned by

$$\{1,\ t,\ \cos(\omega t),\ \sin(\omega t)\}.$$

   Then, compute the limit as ωh tends to 0, being h the stepsize, of the coefficients of the obtained method and check if they fulfill the set of order conditions described in Theorem 3.2 up to a certain integer p.
6. As regards Example 3.9, find the values of β_2 such that third order methods are zero-stable.
7. Write a code in your favorite programming language implementing the three-step method

$$y_{n+3} - \frac{18}{11}y_{n+2} + \frac{9}{11}y_{n+1} - \frac{2}{11}y_n = \frac{6}{11}hf(y_{n+3}, t_{n+3})$$

   for the numerical solution of d-dimensional systems (1.1). You need to recover the two-missing starting values y_1 and y_2, hence you can implement two options: recover both y_1 and y_2 by two consecutive steps of a one-step method; recover y_1 by a one-step method and then y_2 by a two-step method. Provide an experimental evidence of the convergence of the method and of its order.
8. The bound (3.7) estimating the stepsize restriction for the convergence of fixed point iterations in the case implicit LMMs (3.1) is fully computable if the value of the Lipschitz constant L of the vector field is known. Estimating the Lipschitz constant of a function is very important in many fields: for instance, several

estimation algorithms for the Lipschitz constant have been developed in papers on global optimization.

The exercise consists in two parts:

- write a code in your favorite programming language implementing the following estimation algorithm for the Lipschitz constant L of a given scalar function [346].
  Step 1. Compute the approximation P solutions of a scalar initial value problem (1.17) in autonomous form, by a chosen LMM (3.1), corresponding to P different initial values. Denote by $y_n^{i,j}$ the i-th component of the j-th solution in the point $t_n \in \mathcal{I}_h$, $i = 1, 2, \ldots, d$, $j = 1, 2, \ldots, P$. Then, define

$$a_i = \min_{j=1,\ldots,P} \min_{t_n \in \mathcal{I}_h} X_n^{i,j}, \qquad b_i = \max_{j=1,\ldots,P} \max_{t_n \in \mathcal{I}_h} X_n^{i,j},$$

  $i = 1, 2, \ldots, d$.
  Step 2. Generate Q couples of vectors

$$x_k = \left[x_k^1,\ x_k^2,\ \ldots,\ x_k^d\right]^{\mathsf{T}}, \quad y_k = \left[y_k^1,\ y_k^2,\ \ldots,\ y_k^d\right]^{\mathsf{T}},$$

  with $k = 1, 2, \ldots Q$, such that (x_k^i, y_k^i) is uniformly distributed in $[a_i, b_i] \times [a_i, b_i]$, $i = 1, 2, \ldots, d$.
  Step 3. Compute

$$s_k = \frac{|f(x_k) - f(y_k)|^2}{|x_k - y_k|^2}, \qquad k = 1, 2, \ldots, Q.$$

  Step 4. Assume as estimate of L the value of $\max\{s_1, \ldots, s_Q\}$;
- solve the same differential problem by using the implicit Euler method (2.32) and give an experimental confirmation of the sharpness of the bound (3.7), by choosing values of the stepsize above and below this bound (fully computable using the aforementioned estimation algorithm for the Lipschitz constant of the vector field) and checking the convergence of fixed point iterations in correspondence of the chosen values of the stepsize.

9. Using Definition 3.3 of consistency, provide a consistency analysis of Milne-Simpson (3.2) computing its local truncation error and, consequently, infer its order.
10. Develop maximal order convergent LMMs depending on 4 and 5 steps.

Chapter 4
Runge-Kutta Methods

These two papers of Butcher brought elegance and order into the theory of Runge-Kutta methods [. . .] The next sensation came when, «at the Dundee Conference in 1969, a paper by J. Butcher was read which contained a surprising result».

(Gerhard Wanner, Foreword to the book of John C. Butcher [68]. The nested quotation is due to Hans J. Stetter)

As highlighted in Chap. 3, order barriers of linear multistep methods are rather severe. We now move to a different family of methods, i.e., Runge-Kutta methods, enabling better order and stability barriers. The strategy is novel with respect to that beyond LMMs (3.1): indeed, it is no longer of multistep type, as for (3.1), but we move to a multistage strategy relying on the information in some additional points, located inside each subinterval of the domain discretization.

4.1 Genesis and Formulation of Runge-Kutta Methods

Let us consider the integral formulation of (1.1) for $t \in [t_n, t_{n+1}]$, i.e.,

$$y(t) = y(t_n) + \int_{t_n}^{t} f(s, y(s))\mathrm{d}s \tag{4.1}$$

and evaluate it in t_{n+1}, obtaining

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(s, y(s))\mathrm{d}s. \tag{4.2}$$

Let us approximate the integral in the right-hand side by a quadrature formula. We consider s points

$$t_n + c_i h, \quad i = 1, 2, \ldots, s,$$

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_4

being $c_1, c_2, \ldots, c_s$ real numbers, usually belonging to the interval $[0, 1]$, in such a way that $t_n \leq t_n + c_i h \leq t_{n+1}$. This requirements is not mandatory, but it essentially represents a custom in most of the existing methods. Then, we consider the quadrature formula with nodes c_i and weights $b_i \in \mathbb{R}$, $i = 1, 2, \ldots, s$, i.e.,

$$\int_{t_n}^{t_{n+1}} f(s, y(s))\mathrm{d}s \approx h \sum_{i=1}^{s} b_i f(t_n + c_i h, y(t_n + c_i h)). \tag{4.3}$$

Replacing last line in (4.2) leads to

$$y(t_{n+1}) \approx y(t_n) + h \sum_{i=1}^{s} b_i f(t_n + c_i h, y(t_n + c_i h)). \tag{4.4}$$

Last formula has a computational gap: the values of $y(t_n + c_i h)$ are unknown and need to be properly estimated. The trick is always the same: for any $i = 1, 2, \ldots, s$, evaluate (4.1) in $t_n + c_i h$, obtaining

$$y(t_n + c_i h) = y(t_n) + \int_{t_n}^{t_n + c_i h} f(s, y(s))\mathrm{d}s \tag{4.5}$$

and approximate the integral in the right-hand side by a quadrature formula with nodes c_i and weights $a_{ij} \in \mathbb{R}$, $j = 1, 2, \ldots, s$, i.e.,

$$\int_{t_n}^{t_n + c_i h} f(s, y(s))\mathrm{d}s \approx h \sum_{j=1}^{s} a_{ij} f(t_n + c_j h, y(t_n + c_j h)), \quad i = 1, 2, \ldots, s. \tag{4.6}$$

Replacing last line in (4.5) finally gives

$$y(t_n + c_i h) \approx y(t_n) + h \sum_{j=1}^{s} a_{ij} f(t_n + c_j h, y(t_n + c_j h)), \quad i = 1, 2, \ldots, s. \tag{4.7}$$

Equations (4.4) and (4.7) provide approximate equalities involving exact values; let us recast them as exact equalities involving approximate values. By defining

$$Y_i \approx y(t_n + c_i h), \quad i = 1, 2, \ldots, s,$$

we obtain the following relevant family of methods.

Definition 4.1 The family of *Runge-Kutta methods* (RK methods) with respect to the discretization (2.1) is defined by

$$\begin{aligned} y_{n+1} &= y_n + h\sum_{i=1}^{s} b_i f(t_n + c_i h, Y_i), \\ Y_i &= y_n + h\sum_{j=1}^{s} a_{ij} f(t_n + c_j h, Y_j), \quad i = 1, 2, \ldots, s. \end{aligned} \tag{4.8}$$

RK methods (4.8) are characterized by the *weights* b_i, the *nodes* c_i and the scalars a_{ij}, $i, j = 1, 2, \ldots, s$. These are characteristic elements that uniquely define a Runge-Kutta method and, for this reason, they are used to provide the following compact representation. Indeed, let us collect these objects in the following vectors and matrix

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_s \end{bmatrix}, \quad c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_s \end{bmatrix}, \quad A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{ss} \end{bmatrix}.$$

Then, RK methods have a standard representation given by the following array

$$\begin{array}{c|c} c & A \\ \hline & b^{\intercal} \end{array} \tag{4.9}$$

well-known as Butcher tableau, in honor to John C. Butcher (Auckland, 1933), Emeritus Professor at the University of Auckland, well-known as one of the pioneers of the numerical discretization of ODEs, who has given many relevant foundational contributions in establishing a theory of Runge-Kutta methods, as we will discuss later. It is normally assumed that the entries of the vector c of the nodes satisfy the so-called *row-sum condition*, i.e.,

$$c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, 2, \ldots, s, \tag{4.10}$$

i.e., each c_i is the sum of the entries of the i-th row of the matrix A. This is a condition of consistency of the internal stages. Indeed, consider problem (1.1) with

vector field f identically equal to 1 and $y_0 = 0$. Then, its solution is $y(t) = t - t_0$. Correspondingly, the equation for the internal stages (4.8) is given by

$$Y_i = y_n + h \sum_{j=1}^{s} a_{ij}, \quad i = 1, 2, \ldots, s.$$

In hypothesis of consistency and since the solution is a linear polynomial, we obtain

$$t_n + c_i h = t_n + h \sum_{j=1}^{s} a_{ij}, \quad i = 1, 2, \ldots, s,$$

that reduces to (4.10).

Clearly, the vectors and matrix in the Butcher tableau (4.9) are useful to provide a matrix-vector representation of RK methods (4.8), assuming the form

$$\begin{aligned} y_{n+1} &= y_n + h(b^{\intercal} \otimes I) F(Y), \\ Y &= (e \otimes I) y_n + h(A \otimes I) F(Y), \end{aligned} \tag{4.11}$$

where $\otimes$ denotes the standard Kronecker tensor product, I is the identity matrix in $\mathbb{R}^{d \times d}$ and

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{bmatrix} \in \mathbb{R}^{sd}, \quad F(Y) = \begin{bmatrix} f(t_n + c_1 h, Y_1) \\ f(t_n + c_2 h, Y_2) \\ \vdots \\ f(t_n + c_s h, Y_s) \end{bmatrix} \in \mathbb{R}^{sd}, \quad e = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbb{R}^{s}.$$

The vector Y is also known as vector of the *internal stages*. In summary, RK methods (4.8) are one-step formulae for which a single step from t_n to t_{n+1} does not only require the knowledge of y_n, but also the computation of the internal stages by the formula

$$Y = (e \otimes I) y_n + (A \otimes I) F(Y). \tag{4.12}$$

As a consequence, the computational cost needed to compute the vector Y is strongly dependent on the structure of the matrix A. Indeed:

- if A is strictly lower triangular, (4.12) is an explicit formula for the calculation of Y and the corresponding RK method is said to be *explicit*;
- if A is a lower triangular matrix, (4.12) is a structured system of nonlinear algebraic equations that has to be solved at each step and the corresponding RK method is said to be *diagonally-implicit*. This class of RK methods is not covered

in this book, but the reader can see, for instance, to [7, 53, 62, 198, 221, 242] and references therein;
- if A is a full matrix, (4.12) is a full nonlinear system of algebraic equations to be solved at each step and the corresponding RK method is said to be *implicit*.

Let us now give some historical notes. As the name clearly states, the formulation of Runge-Kutta methods is due to Carl Runge and Willem Kutta: they were formerly introduced by Runge in 1895, formulated as an extension of the Euler method (2.19) able to achieve higher accuracy; the idea was then extended by Kutta in 1901, leading to the formulation nowadays used for RK methods. Let us give a portrait of Runge and Kutta, based on the information reported in the gifted MacTutor History of Mathematics Archive (https://mathshistory.st-andrews.ac.uk/Biographies/Kutta/, https://mathshistory.st-andrews.ac.uk/Biographies/Runge/) and in the celebrative paper [72] by John C. Butcher and Gerhard Wanner.

A Portrait of Carl David Tolmé Runge

Runge was born in Bremen in 1856, but he spent his early years in Havana, with his parents, three brothers and four sisters. Both his father Julius and his mother Fanny Tolmé belonged to a family of merchants. Fanny was the daughter of an English merchant, even if her family was of French descent, and she used to speak English with Julius and her children who grew up with English as first language. After his retirement, Julius and his family moved back to Bremen, both he died soon.

Carl Runge attended completed his high school studies in Bremen in 1875 and, after spending six months with his mother visiting Italy, he enrolled at the University of Munich at Easter 1876 to first study literature and philosophy, before moving soon to mathematics and physics. During his studies, Max Planck was his fellow student and they became close friends.

In 1877 Planck and Runge moved to Berlin; Runge was really impressed by the lectures of Karl Weierstrass, who will become his Ph.D. advisor, and decided to turn to pure mathematics. He got his Ph.D. in 1880, discussing a thesis on differential geometry entitled "Über die Krümmung, Torsion und geodä tische Krümmung der auf einer Flä che gezogenen Curven" (About the curvature, torsion and geodesic curvature of the curves drawn on a surface); such a topic came out from several discussions with other students in the Mathematischer Verein where he was an active member, rather than with his advisor. In Berlin he also became an impressive ice skater and had a very intense social life.

In 1881, after qualifying for the habilitation as Gymnasium professor, he started his collaboration with Kronecker. He designed a procedure for the numerical solution of algebraic equations, included in his Habilitation thesis completed in 1883 in Berlin, where he continued to do research on algebra and

(continued)

function theory in the group of mathematicians built up around Kronecker. In his research career the visit to Mittag-Leffler in Stockholm in 1884 was absolutely crucial: after that Runge wrote several papers published in Acta Mathematica in 1885.

In the same year, Runge engaged with Aimé du Bois-Reymond but her father (Émile, a professor friend of Carl), influenced by his strict views due to his Pietist formation, prohibited them to get married until Carl had achieved a professor position. This achievement occurred in 1886, when Runge got a chair at the Technische Hochschule of Hannover. They finally married in 1887 and remained in Hannover for 18 years. His colleague Friedrich Paschen gave a nice portrait of Runge and his family in the obituary appeared in Astrophysical Journal 69, 317–321 (1929):

> They had four daughters and two sons, one of whom was killed in the war [World War I]. Runge's home at Hannover [...] will never be forgotten by those who had the privilege of entering it. The family cultivated many sciences and arts. Runge himself played the piano, and he and his children would often render musical classics such as the 'Matthäus Passion'. Runge was a man of affairs and of great personal charm. He was fond of all kinds of sports and practiced bicycling, gymnastics, and swimming. At Hannover he used to ride his bicycle a distance of about eight kilometers from his house top the Technische Hochschule four times a day. In all his activities he placed scientific things foremost and was willing to sacrifice everything to their advancement.

After achieving his professorship, Runge moved from pure mathematics to Physics, establishing a fruitful collaboration with Heinrich Kayser on spectroscopy for seven years, until Kayser left Hannover in 1894 (he moved to University of Bonn to cover the professor position left vacant by Heinrich Hertz, who died at age 36 because of blood poisoning). Then, Carl Runge started a collaboration with Friedrich Paschen, an experimentalist, and they worked together at Hannover for seven years. In 1895 he published his famous paper *"Über die numerische Auflösung von Differentialgleichungen"* (About the numerical resolution of differential equations) [304] giving rise to this first contribution on what would have become the widely studied Runge-Kutta methods.

Runge had several visits abroad: in England in 1895 he got in touch with Lord Rayleigh; in the United States in 1897 he became friend with Michelson. He got a professorship proposal in the United States, but he declined. In 1901 Paschen left Hannover and moved to the University of Tübingen. Runge continued his work with Julius Precht who had achieved the Extraordinary Chair of Theoretical Physics at Heidelberg. In 1901 he published another famous paper *"Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten"* (On empirical functions and the interpolation with equidistant ordinates), where he described the phenomenon occurring in

(continued)

polynomial interpolation with many equidistant nodes, now known as Runge phenomenon.

Felix Klein was able to let Carl Runge being professor of Applied Mathematics in Göttingen in 1904, where he remained until his retirement in 1925. At this stage, Runge gave his contributions on many numerical and graphical methods and in Göttingen was a very influential figure. He is acknowledged as a pioneer in introducing this kind of mathematics in Germany. He gave lectures on graphical methods at Columbia University in New York from October 1909 to January 1910.

Runge retired in 1923, but he still continued his activity in Göttingen until the arrival of his successor Gustav Herglotz, in 1925. Six months after his 70th birthday he died of a heart attack.

A Portrait of Martin Wilhelm Kutta

Kutta was born in 1867 in Pitschen (now Byczyna, in Poland). He has tragically lost his parents when he was very young, so he was educated with his brother Karl by an uncle in Breslau, where he attended the Gymnasium and the University, from 1885 to 1890. Then he moved to Munich where he studied from 1891 to 1894. Within his broad interests, mathematics occupied a central role, but also languages, music and art for all his life.

He got a position in the Technische Hochschule of Munich as assistant in mathematics and physics from 1894. He achieved his Ph.D. at the University of Munich for his thesis *"Beiträge zur näherungsweisen Integration totaler Differentialgleichungen"* (Contributions for the approximate integration of total differential equations) in 1900, advised by Lindemann and Bauer. The thesis, published a year later, contains the famous Runge-Kutta method for the discretization of ordinary differential equations, according to the notation nowadays adopted.

He was inspired by his colleague Sebastian Finsterwalder, who brought photographs of an early aircraft to the Institute and led Kutta interests to aerodynamics. This was the topic of his habilitation thesis (containing the relevant Zhukovsky-Kutta theorem describing the lift on an aerofoil) submitted in 1902, after which he got a promotion as extraordinary professor in Applied Mathematics in 1907. In 1909 he moved to the University of Jena and in 1910 he was achieved the position of ordinary professor in the Technische Hochschule at Aachen. In 1911 he moved to Stuttgart, where he remained, until his retirement in 1935, and focused on teaching to engineers who got a huge benefit from his inspiring presentation.

(continued)

Finsterwalder inspired also the interests of Kutta in glaciers: he made measurements of glaciers through photographs of the East Alps, as well as he worked on mapping the area covered by glaciers. He was also interested in history of mathematics, as well as in historical literature, which he was able to cultivate attending the active seminar of the Technische Hochschule in Munich. He died in 1944 in Fürstenfeldbruck. His colleague Pfeiffer describes Kutta as person with very broad interests, but also very lonely; he writes:

"I had the good fortune in my life to become acquainted with a large number of outstanding mathematicians [...], but I never met a mathematician who had such a deep interest and familiarity with so many different areas of mental activity as Kutta".

Runge-Kutta methods are one-step methods whose incremental function is the vector field of the continuous problem (1.1), that is Lipschitz continuous to ensure Hadamard well-posedness. Then, in force of a natural extension of Theorem 2.5 for implicit methods as well, all RK methods are zero-stable. Moreover, by Lax equivalence theorem, all RK methods of order greater or equal than 1 are convergent. Next section provides a very elegant analysis of order, fruit of the talent of John C. Butcher.

4.2 Butcher Theory of Order

Butcher theory for the analysis of the order of Runge-Kutta methods relies on tools that match numerical analysis, graph theory, differentiation of vector fields, and so on. Later, its meaningful relationship with other topics, such as group theory [194], as well as quantum field theory [94] has been discovered. A featured monograph on Butcher theory of order and related issues is authored by John C. Butcher himself [68], reviewed in [116].

The basic tool that connects all the aforementioned fields is the notion of rooted trees. The following presentation is far from being an exhaustive description of graph theory: here we only explain the basic notions we need to understand Butcher theory.

4.2.1 Rooted Trees

The set of rooted trees $\mathbf{T}$ can be graphically represented as follows

$$\mathbf{T} = \left\{ \bullet, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \vcenter{\hbox{\includegraphics[height=1.2em]{}}}, \; \ldots \right\},$$

i.e., it is the set of trees where a special node is highlighted, the *root*, above given by the bottom node of each tree. However, a more practical representation of rooted trees can be given in terms of the so-called *square bracket notation*, as follows:

- denote the rooted tree with a single node by τ;
- if t_1 is the subtree originated by cutting the root of a tree $t \in \mathbf{T}$, then we use the notation $t = [t_1]$. In other terms, the square brackets denote the operation of removing the root from a given tree.

This notation leads to the following alternative representation of the set of rooted trees

$$\mathbf{T} = \left\{ \tau \ , \ [\tau] \ , \ [\tau^2] \ , \ [[\tau]] \ , \ [\tau^3] \ , \ [\tau[\tau]] \ , \ [[\tau^2]] \ , \ [[[\tau]]] \ , \ \ldots \right\}.$$

Three basic functions on $\mathbf{T}$ are used in Butcher theory. The first one is

$$\rho : \mathbf{T} \to \mathbb{R}$$

defined as *order* of a rooted tree; it is the number of the nodes in a given tree. For instance, $\rho([[\tau]]) = 3$. The second function

$$\sigma : \mathbf{T} \to \mathbb{R}$$

is denoted as the *symmetry* of a rooted tree and represents the cardinality of its automorphism group. Such a function is recursively computed by the formula

$$\sigma(\tau) = 1,$$
$$\sigma(t) = \prod_{i=1}^{k} m_i! \sigma(t_i)^{m_i},$$

supposing that $t = [t_1^{m_1} \ t_2^{m_2} \ \ldots \ t_k^{m_k}]$. For instance, $\sigma([[\tau]]) = \sigma([\tau]) = \sigma(\tau) = 1$. Finally, the function

$$\gamma : \mathbf{T} \to \mathbb{R}$$

denoted as the *density* of a rooted tree is defined by the recursion

$$\gamma(\tau) = 1,$$
$$\gamma(t) = \rho(t) \prod_{i=1}^{k} \gamma(t_i)^{m_i}.$$

Table 4.1 Square bracket notation, symmetry and density for rooted trees of order up to 4

t	•							
Square bracket notation	τ	$[\tau]$	$[\tau^2]$	$[[\tau]]$	$[\tau^3]$	$[\tau[\tau]]$	$[[\tau^2]]$	$[[[\tau]]]$
$\rho(t)$	1	2	3	3	4	4	4	4
$\sigma(t)$	1	1	2	1	6	1	2	1
$\gamma(t)$	1	2	3	6	4	8	12	24

For instance, $\gamma([[\tau]]) = 3\gamma([\tau]) = 6\gamma(\tau) = 6$. Table 4.1 lists the trees of order up 4, their square bracket representations, their symmetries and densities. A further example of calculation is given in the example below.

Example 4.1 Consider the rooted tree t graphically represented by

admitting the square brackets notation $t = [[[\tau]^2]]$. The order of this rooted tree is $\rho(t) = 6$. Its symmetry is given by

$$\sigma(t) = \sigma([[\tau]^2]) = 2\sigma([\tau])^2 = 2\sigma(\tau)^2 = 2,$$

while its density is

$$\gamma(t) = 6\gamma([[\tau]^2]) = 30\gamma([\tau])^2 = 120.$$

There is a deep connection between rooted trees and differentiation of vector fields. Such a link is at the basis of Butcher theory and its precursors, well described in the paper [265]. The first precursor was Robert Henry Merson (1921–1992), a scientist at the Royal Aircraft Establishment in the United Kingdom who became popular for his involvement in the computations of an accurate orbit for Sputnik 1, whose launch occurred in 1957. In the same year he was invited in Salisbury, South Australia, to give a plenary talk during a conference on Automatic Computing Machines. In his paper [266] he described the one-to-one correspondence among derivatives and rooted trees, even if the full theory will only be completed later by

John Butcher, who attended the talk by Merson in 1957: as the authors of [265] properly state, *"the seed (of Butcher theory) was planted there"*. But exactly one century before, Caley [80] introduced trees with the same purpose as in Butcher theory, i.e., understanding and effectively representing the interaction of vector fields repeatedly applied to one another, and for one century this aspect was totally forgotten by the literature and reconsidered (actually, more or less from scratch) only when the theory of numerical methods was established with more rigor, in the second half of twentieth century.

4.2.2 Elementary Differentials

Let us now analyze the connection between rooted trees and differentiation of vector fields. For our convenience, we consider an autonomous differential problem

$$y'(x) = f(y(x)),$$

where $f : \mathbb{R}^d \to \mathbb{R}^d$. Only in this subsection we denote the independent variable by x, in order to avoid confusions with the symbol $t \in \mathbf{T}$ denoting a rooted tree. Moreover, we omit the dependency on x in order to make the notation less arduous to follow as possible. Now, taking into account that $y' = f(y)$ let us compute the derivative with respect to x side by side:

$$y'' = \frac{\mathrm{d}}{\mathrm{d}x} f(y) = f'(y)y' = f'(y)f(y),$$

where $f'(y)$ is Jacobian matrix containing all the partial derivatives of f, so it is a linear operator. Let us differentiate again:

$$y''' = \frac{\mathrm{d}}{\mathrm{d}x} \big(f'(y)f(y)\big) = f''(y)(f(y), f(y)) + f'(y)f'(y)f(y).$$

Observe that the second derivative of the vector field is a bilinear form, whose i-th component is given by

$$\sum_{j,k=1}^{d} \frac{\partial^2 f_i(y)}{\partial y_j \partial y_k} f_j(y) f_k(y).$$

Let us also give the expression of the fourth derivative

$$\begin{aligned} y^{(iv)} &= \frac{\mathrm{d}}{\mathrm{d}x}\left(f''(y)(f(y),f(y)) + f'(y)f'(y)f(y)\right) \\ &= f'(y)f'(y)f'(y)f(y) + f(y)f''(y)(f(y),f(y)) \\ &\quad + 3f''(y)(f'(y)f(y),f(y)) + f'''(y)(f(y),f(y),f(y)), \end{aligned}$$

where the third derivative of f is the multilinear operator having, as general i-th component

$$\sum_{j,k,\ell=1}^{d} \frac{\partial^3 f_i(y)}{\partial y_j \partial y_k \partial y_\ell} f_j(y) f_k(y) f_\ell(y).$$

More in general, the derivative of order k is the multilinear operator $f^{(k)}(y)(z^1, z^2, \ldots, z^k)$, being $z^1, z^2, \ldots, z^k \in \mathbb{R}^d$ its k arguments, having the form

$$\sum_{i_1=1}^{d}\sum_{i_2=1}^{d}\cdots\sum_{i_k=1}^{d} \frac{\partial^k f_i(y)}{\partial y_{i_1}^1 \partial y_{i_2}^2 \cdots \partial y_{i_k}^k} z_{i_1}^1 z_{i_2}^2 \cdots z_{i_k}^k.$$

When the author of this book attended the main lecture that John Butcher gave in 2008 in Auckland in the occasion of GLADE 2008 conference, honoring his 75th birthday, describing the representation of the derivatives of y given above, Butcher said: "Can you see a tree structure?". Let us collect together above derivatives to appreciate such a tree structure:

$$\begin{aligned} y' &= f(y), \\ y'' &= f'(y)f(y), \\ y''' &= f''(y)(f(y),f(y)) + f'(y)f'(y)f(y), \\ y^{(iv)} &= f'(y)f'(y)f'(y)f(y) + f(y)f''(y)(f(y),f(y)) \\ &\quad + 3f''(y)(f'(y)f(y),f(y)) + f'''(y)(f(y),f(y),f(y)). \end{aligned} \tag{4.13}$$

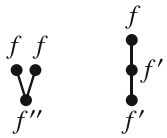
Each summand appearing in the right-hand sides of (4.13) is called *elementary differential* and will be defined below with more rigor. The first thing we can observe is that the number of elementary differentials appearing in the derivative of order k equals the number of rooted trees of order k in $\mathbf{T}$. But actually, the link is much deeper: there is one-to-one connection among rooted trees and elementary differentials. In order to appreciate this issue, let us design the following labelling:

- a leaf in a rooted tree is labelled by f;
- a node having k children is labelled by $f^{(k)}$.

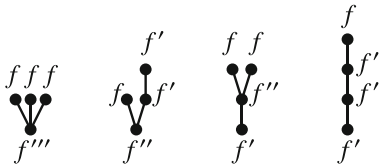
This leads to the following labelling:

for the tree τ of order 1;

for the tree $[\tau]$ of order 2;



for the trees $[\tau^2]$ and $[[\tau]]$ of order 3;



for the trees $[\tau^3]$, $[\tau[\tau]]$, $[[\tau^2]]$ and $[[[\tau]]]$ of order 4. We can observe the perfect match among above trees and the corresponding elementary differentials in (4.13), also listed in Table 4.2.

We also adopt the following rigorous definition of elementary differential as given by Butcher [67].

Definition 4.2 For a given rooted tree $t = [t_1\ t_2\ \ldots\ t_m] \in \mathbf{T}$ and a function $f : \mathbb{R}^d \to \mathbb{R}^d$ analytic in a neighborhood of y, the *elementary differential* $F(t)(y)$ is defined by

$$F(t)(y) = f^{(m)}(F(t_1)(y), F(t_2)(y), \ldots, F(t_m)(y)),$$

assuming that $F(\tau)(y) = f(y)$.

Table 4.2 Elementary differentials and associated rooted trees up to order 4

t	Elementary differential
	$f(y)$
	$f'(y)f(y)$
	$f''(y)(f(y),f(y))$
	$f'(y)f'(y)f(y)$
	$f'''(y)(f(y),f(y),f(y))$
	$f''(y)(f'(y)f(y),f(y))$
	$f'(y)f''(y)(f(y),f(y))$
	$f'(y)f'(y)f'(y)f(y)$

Example 4.2 Consider again the rooted tree $t = [[[\tau]^2]]$, introduced in Example 4.1. The corresponding elementary differential is

$$\begin{aligned} F(t)(y) &= f'(y)F([[\tau]^2]) \\ &= f'(y)f''(y)(F([\tau])(y), F([\tau])(y)) \\ &= f'(y)f''(y)(f'(y)F(\tau)(y), f'(y)F(\tau)(y)) \\ &= f'(y)f''(y)(f'(y)f(y), f'(y)f(y)). \end{aligned}$$

4.2.3 B-Series

Elementary differentials also give an alternative way to represent the Taylor expansion of the exact solution of a differential problem, i.e.,

$$y(x+h) = y(x) + \sum_{k=1}^{\infty} \frac{h^k}{k!} y^{(k)}(x).$$

Indeed, elementary differentials are useful to provide the following alternative expansion with respect to rooted trees, i.e.,

$$y(x+h) = y(x) + \sum_{t \in \mathbf{T}} h^{\rho(t)} \alpha(t) F(t)(y(x)),$$

depending on the unknown coefficients $\alpha(t)$, $t \in \mathbf{T}$, we are now ready to compute. Such an expression is denoted in the literature as *Butcher-series* (in short, *B-series*)

of the exact solution. This celebrative name belongs to Ernst Hairer (Nauders, 1949) and Gerhard Wanner (Innsbruck, 1942), professors at the University of Geneva, eminent pioneers for Numerical Analysis. Their contact with John Butcher is dated back to the early 1970s of the twentieth century and gets its origin from the interest that the paper [60] had arisen in Wanner at that time, when he was a 28-years old professor in Innsbruck. University of Innsbruck was celebrating its 300th anniversary of the foundation and each professor had to possibility to invite a guest lecturer. Wolfgang Gröbner, who was Wanner's professor, got from Gerhard the suggestion to invite John Butcher as lecturer. The lecture had a special attendant in the audience: the young Ernst Hairer, who was the best student of Wanner's course the year before. That invitation was the fundamental seed that led Hairer and Wanner introduce the notion of Butcher series and Butcher group in 1974 [194].

In [55], Butcher gave the complete expression of the B-series of the exact solution of a differential problem, explained in Theorem 4.1 relying on the following lemma, whose proof is here omitted (the reader can find it, for instance, in [67]).

Lemma 4.1 *For a given function* $\theta : \mathbf{T} \to \mathbb{R}$,

$$hf\left(y_0 + \sum_{t\in\mathbf{T}} \theta(t)\frac{h^{\rho(t)}}{\sigma(t)}F(t)(y_0)\right) = \sum_{t\in\mathbf{T}} \widetilde{\theta}(t)\frac{h^{\rho(t)}}{\sigma(t)}F(t)(y_0),$$

where

$$\widetilde{\theta}(t) = \begin{cases} 1, & t = \tau, \\ \displaystyle\prod_{i=1}^{m} \theta(t_i), & t = [t_1\ t_2\ \ldots\ t_m]. \end{cases}$$

Theorem 4.1 (B-Series of the Exact Solution) *The B-series of the exact solution in* $x_0 + h$ *of the initial value problem*

$$y(x_0 + h) = y_0 + \int_{x_0}^{x_0+h} f(y(s))ds \tag{4.14}$$

is given by

$$y(x_0 + h) = y_0 + \sum_{t\in\mathbf{T}} \frac{h^{\rho(t)}}{\sigma(t)\gamma(t)}F(t)(y_0). \tag{4.15}$$

Proof Following [67], we give a proof based on Picard iterations (1.9). To this purpose, we evaluate (4.14) as follows

$$y(x_0 + h\xi) = y_0 + \int_{x_0}^{x_0+h\xi} f(y(s))\mathrm{d}s, \quad \xi \in [0, 1]$$

and perform a change of variable $s = x_0 + h\xi$, obtaining

$$y(x_0 + h\xi) = y_0 + h\int_0^{\xi} f(y(x_0 + h\xi))\mathrm{d}\xi.$$

Let us associate Picard iterations to this problem

$$y_n(x_0 + h\xi) = y_0 + h\int_0^{\xi} f(y_{n-1}(x_0 + h\xi))\mathrm{d}\xi, \quad n \geq 1 \tag{4.16}$$

and let us prove that a generic Picard iteration satisfies

$$y_n(x_0 + h\xi) = y_0 + \sum_{t\in\mathbf{T}_n} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\gamma(t)} F(t)(y_0) + O(h^{n+1}), \tag{4.17}$$

being $\mathbf{T}_n$ the set of rooted trees of order up to n. The proof is given by induction. The case $n = 1$ is obvious. Let us assume that formula (4.17) is true for $n - 1$ and replace this assumption in (4.16), obtaining

$$y_n(x_0 + h\xi) = y_0 + h\int_0^{\xi} f\left(y_0 + \sum_{t\in\mathbf{T}_{n-1}} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\gamma(t)} F(t)(y_0)\right)\mathrm{d}\xi + O(h^n).$$

Let us apply Lemma 4.1 on the last integrand, assuming

$$\theta(t) = \frac{1}{\gamma(t)}$$

and leading to

$$(h\xi) f\left(y_0 + \sum_{t\in\mathbf{T}_{n-1}} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\gamma(t)} F(t)(y_0)\right) = \sum_{t\in\mathbf{T}_{n-1}} \widetilde{\theta}(t) \frac{(h\xi)^{\rho(t)}}{\sigma(t)} F(t)(y_0),$$

with

$$\widetilde{\theta}(t) = \frac{1}{\prod_{i=1}^{m} \gamma(t_i)},$$

being $t = [t_1\ t_2\ \ldots\ t_m]$. Therefore,

$$\begin{aligned}
y_n(x_0 + h\xi) &= y_0 + h \int_0^{\xi} f\left(y_0 + \sum_{t \in \mathbf{T}_{n-1}} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\gamma(t)} F(t)(y_0) \right) \mathrm{d}\xi + O(h^n) \\
&= y_0 + \int_0^{\xi} \sum_{t \in \mathbf{T}_{n-1}} \widetilde{\theta}(t) \frac{h^{\rho(t)} \xi^{\rho(t)-1}}{\sigma(t)} F(t)(y_0) \mathrm{d}\xi + O(h^n) \\
&= y_0 + \sum_{t \in \mathbf{T}_{n-1}} \frac{h^{\rho(t)}}{\sigma(t) \prod_{i=1}^{m} \gamma(t_i)} F(t)(y_0) \int_0^{\xi} \xi^{\rho(t)-1} \mathrm{d}\xi + O(h^n) \\
&= y_0 + \sum_{t \in \mathbf{T}_{n-1}} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\rho(t) \prod_{i=1}^{m} \gamma(t_i)} F(t)(y_0) + O(h^n) \\
&= y_0 + \sum_{t \in \mathbf{T}_n} \frac{(h\xi)^{\rho(t)}}{\sigma(t)\gamma(t)} F(t)(y_0) + O(h^{n+1}).
\end{aligned}$$

The limit as n goes to infinity, in force of the convergence of Picard iterations proved in Theorem 1.3, gives the thesis. □

4.2.4 *Elementary Weights*

With analogous arguments as in Theorem 4.1, we can compute the B-series of the numerical solution computed by (4.8). Such a computation relies on an additional tool, associated to the coefficients of the RK method according to the following definition.

Definition 4.3 For a given rooted tree $t = [t_1\ t_2\ \ldots\ t_m] \in \mathbf{T}$, the *derivative weights* $(\Phi_i D)(t)$, *internal weights* $\Phi_i(t)$ and *external weights* $\Phi(t)$ associated to a RK method (4.8) are recursively defined, for $i = 1, 2, \ldots, s$, by

$$
\begin{aligned}
(\Phi_i D)(\tau) &= 1, \\
\Phi_i(t) &= \sum_{j=1}^{s} a_{ij} (\Phi_j D)(t), \\
(\Phi_i D)(t) &= \prod_{j=1}^{m} \Phi_i(t_j), \\
\Phi(t) &= \sum_{i=1}^{s} b_i (\Phi_i D)(t).
\end{aligned}
$$

As in the case of elementary differentials, there is a one-to-one connection among rooted trees and external elementary weights as well, described by the following procedure:

- label each node of a given rooted tree;
- if i is the label chosen for the root, it is associated to b_i;
- a couple of nodes labelled by i and j and connected by an edge of the tree is associated to a_{ij};
- sum over all the indices employed in the labelling, using the row-sum condition (4.10), when possible.

Then,

- for the tree τ, labelled as follows

the associated external elementary weight is given by

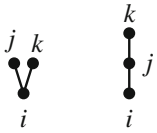
$$\Phi(\tau) = \sum_{i=1}^{s} b_i;$$

- for the tree $[\tau]$, labelled by

the corresponding external elementary weight is

$$\Phi([\tau]) = \sum_{i,j=1}^{s} b_i a_{ij} = \sum_{i=1}^{s} b_i c_i;$$

- for the trees $[\tau^2]$ and $[[\tau]]$ of order 3, labelled as follows

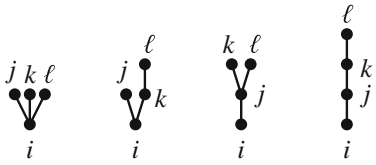


the associated external elementary weights are

$$\Phi([\tau^2]) = \sum_{i,j,k=1}^{s} b_i a_{ij} a_{ik} = \sum_{i,k=1}^{s} b_i c_i a_{ik} = \sum_{i=1}^{s} b_i c_i^2,$$

$$\Phi([[\tau]]) = \sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} = \sum_{i,j=1}^{s} b_i a_{ij} c_j;$$

- for the trees $[\tau^3]$, $[\tau[\tau]]$, $[[\tau^2]]$ and $[[[\tau]]]$ of order 4, labelled by



the corresponding external elementary weights are

$$\Phi([\tau^3]) = \sum_{i,j,k,\ell=1}^{s} b_i a_{ij} a_{ik} a_{i\ell} = \sum_{i=1}^{s} b_i c_i^3,$$

$$\Phi([\tau[\tau]]) = \sum_{i,j,k,\ell=1}^{s} b_i a_{ij} a_{ik} a_{k\ell} = \sum_{i,k=1}^{s} b_i c_i a_{ik} c_k;$$

$$\Phi([[\tau^2]]) = \sum_{i,j,k,\ell=1}^{s} b_i a_{ij} a_{jk} a_{j\ell} = \sum_{i,j=1}^{s} b_i a_{ij} c_j^2;$$

$$\Phi([[[\tau]]]) = \sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} a_{k\ell} = \sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k.$$

Hence, due to the row-sum assumption (4.10), each leaf of a rooted tree corresponds to the node c having as subscript the same index of the original parent. In summary, the external elementary weights associated to the trees of orders up to 4 are listed in Table 4.3.

Table 4.3 External elementary weights and associated rooted trees up to order 4

t	External elementary weights
	$\sum_{i=1}^{s} b_i$
	$\sum_{i=1}^{s} b_i c_i$
	$\sum_{i=1}^{s} b_i c_i^2$
	$\sum_{i,j=1}^{s} b_i a_{ij} c_j$
	$\sum_{i=1}^{s} b_i c_i^3$
	$\sum_{i,k=1}^{s} b_i c_i a_{ik} c_k$
	$\sum_{i,j=1}^{s} b_i a_{ij} c_j^2$
	$\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k$

Example 4.3 Let us consider again the rooted tree $t = [[[\tau]^2]]$ and compute the corresponding external elementary weight. We first recursively proceed by applying Definition 4.3:

$$\begin{aligned}
\Phi(t) &= \sum_{i=1}^{s} b_i \left(\Phi_i D\right)\left([[[\tau]^2]]\right) = \sum_{i=1}^{s} b_i \Phi_i\left([[\tau]^2]\right) \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\Phi_j D\right)\left([[\tau]^2]\right) = \sum_{i,j=1}^{s} b_i a_{ij} \Phi_j([\tau])^2 \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\sum_{\ell=1}^{s} a_{j\ell} \left(\Phi_\ell D\right)([\tau])\right)^2 \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\sum_{\ell=1}^{s} a_{j\ell} \Phi_\ell(\tau)\right)^2 \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\sum_{\ell,m=1}^{s} a_{j\ell} a_{\ell m}\right)^2 \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\sum_{\ell=1}^{s} a_{j\ell} c_\ell\right)^2 .
\end{aligned}$$

Let us now compute $\Phi(t)$ by directly acting on the following graph labelling

obtaining

$$\begin{aligned}
\Phi(t) &= \sum_{i,j,k,\ell,m,n=1}^{s} b_i a_{ij} a_{jk} a_{km} a_{j\ell} a_{\ell n} = \sum_{i,j,k,\ell=1}^{s} b_i a_{ij} a_{jk} c_k a_{j\ell} c_\ell \\
&= \sum_{i,j=1}^{s} b_i a_{ij} \left(\sum_{\ell=1}^{s} a_{j\ell} c_\ell\right)^2 .
\end{aligned}$$

We can now state the following result on the B-series of the numerical solution computed by RK methods (4.8). The proof, similar to that of Theorem 4.1, is here omitted, but the interested reader can find it in [67].

Theorem 4.2 (B-Series of the Numerical Solution) *The B-series of the numerical solution given by a single step of the RK method* (4.8) *for the computation of* y_1*, given the initial value* y_0*, assumes the form*

$$y_1 = y_0 + \sum_{t \in \mathbf{T}} \frac{1}{\sigma(t)} \Phi(t) h^{\rho(t)} F(t)(y_0). \tag{4.18}$$

4.2.5 Order Conditions

The results developed so far, that led us to the definition of B-series for both the exact solution of (1.1) and its approximation computed by a given RK-method (4.8), guide us toward the following result, elegantly and effectively giving the set of order conditions for these methods.

Theorem 4.3 (Butcher) *A given RK method* (4.8) *has order* p *if and only if*

$$\Phi(t) = \frac{1}{\gamma(t)},$$

for any $t \in \mathbf{T}$ *of order* $\rho(t) \leq p$.

Proof The thesis holds true from the direct comparison of the B-series of the exact solution (4.15) and that of the numerical solution (4.18). □

As a consequence of Theorem 4.3, we can give a criterion of convergence of RK methods based on a straightforward calculation.

Table 4.4 Order conditions up to 4 for Runge-Kutta methods (4.8)

Order	Order conditions
1	$\sum_{i=1}^{s} b_i = 1$
2	$\sum_{i=1}^{s} b_i c_i = \frac{1}{2}$
3	$\sum_{i=1}^{s} b_i c_i^2 = \frac{1}{3}$ $\sum_{i,j=1}^{s} b_i a_{ij} c_j = \frac{1}{6}$
4	$\sum_{i=1}^{s} b_i c_i^3 = \frac{1}{4}$ $\sum_{i,k=1}^{s} b_i c_i a_{ik} c_k = \frac{1}{8}$ $\sum_{i,j=1}^{s} b_i a_{ij} c_j^2 = \frac{1}{12}$ $\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k = \frac{1}{24}$

Corollary 4.1 *A given RK method (4.8) is convergent if and only if*

$$\sum_{i=1}^{s} b_i = 1. \tag{4.19}$$

Proof Since RK methods are all zero-stable, according to Theorem 2.6, consistent RK methods are also convergent. Consistency requires achieving at least order 1 that means, according to Theorem 4.3

$$\Phi(\tau) = \frac{1}{\gamma(\tau)},$$

equivalent to condition (4.19). □

Table 4.4 collects the set of algebraic conditions that the coefficients of a given RK method (4.8) have to satisfy in order to achieve order up to 4.

4.3 Explicit Methods

We now aim to analyze explicit RK methods, whose Butcher tableau (4.9) is given by

$$\begin{array}{c|cccc} c_1 & & & & \\ c_2 & a_{21} & & & \\ \vdots & \vdots & \ddots & & \\ c_s & a_{s1} & \dots & a_{s,s-1} & \\ \hline & b_1 & \dots & b_{s-1} & b_s \end{array}$$

i.e., the matrix A is strictly lower triangular. We observe that the zero entries of the matrix A are normally not reported in the Butcher tableau. As a consequence of this structure for the matrix A, the internal stages of the method can be explicitly calculated. Indeed, by the formula for the internal stages given in (4.8), we have

$$Y_i = y_n + h \sum_{j=1}^{i-1} a_{ij} f(t_n + c_j h, Y_j), \quad i = 1, 2, \dots, s,$$

so the i-th stage depends on the previous $i - 1$ stages. The strictly lower triangular structure of the matrix A is certainly useful because it makes the implementation of the corresponding methods easier, but it also has a nontrivial drawback, which has been highlighted by Butcher in [59]. The fundamental results arising from his analysis of explicit methods are the following.

Theorem 4.4 (Butcher Barrier for Explicit Methods) *The maximum attainable order of an explicit s-stage RK method is $p = s$.*

Theorem 4.5 (Butcher Barrier for Explicit Methods with $s > 5$) *If $s > 5$, no RK methods of order $p = s$ exist.*

The proofs of above results are omitted, but the interested reader can find various proofs in [67, 198, 242]. According to Butcher barriers, we can construct explicit methods of order $p = s$, with $s = 1, 2, 3, 4$. Let us consider each case, starting from $s = 1$. One-stage explicit methods satisfying the row-sum condition (4.10) and the

convergence condition (4.19) have the following Butcher tableau

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

and, as a consequence, the only one-stage explicit method is the explicit Euler method (2.19).

Two-stage explicit methods satisfying the row-sum condition (4.10) and the convergence condition (4.19) have the following Butcher tableau

$$\begin{array}{c|cc} 0 & & \\ c_2 & c_2 & \\ \hline & 1-b_2 & b_2 \end{array}$$

so, they depend on two parameters: c_2 and b_2. Their maximum attainable order is $p = s = 2$ and it is reached by imposing the order 2 condition in Table 4.4, i.e., $b_2 c_2 = \frac{1}{2}$. Hence, we obtain the Butcher tableau

$$\begin{array}{c|cc} 0 & & \\ c_2 & c_2 & \\ \hline & 1-\frac{1}{2c_2} & \frac{1}{2c_2} \end{array} \tag{4.20}$$

of a one-parameter family of maximal order methods. A value of c_2 can be selected, for instance, in order to have the best stability properties, as it will be highlighted in Chap. 6. An example of two-stage method of order 2, obtained from (4.20) imposing $c_2 = \frac{1}{2}$, is the so-called *explicit midpoint method*

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

also denoted as *modified Euler method* derived by Heun (for this reason, it is also denoted as *Euler-Heun* method). Another famous method derived by Heun is the

explicit trapezoidal method, obtained from (4.20) with $c_2 = 1$,

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

denoted in the literature also as *improved Euler method*. The choice $c_2 = \frac{2}{3}$ in (4.20) leads to the so-called *Ralston method*

$$\begin{array}{c|cc} 0 & & \\ \frac{2}{3} & \frac{2}{3} & \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

developed by Anthony Ralston (New York City, 1930) in [296] as a method with minimal error constant.

Finally, let us consider three-stage explicit methods satisfying the row-sum condition (4.10) and the convergence condition (4.19), then having the following Butcher tableau

$$\begin{array}{c|ccc} 0 & & & \\ c_2 & c_2 & & \\ c_3 & c_3 - a_{32} & a_{32} & \\ \hline & 1 - b_2 - b_3 & b_2 & b_3 \end{array}$$

so, they depend on five parameters: c_2, c_3, a_{31}, b_2 and b_3. Their maximum attainable order is $p = s = 3$ and it is reached by imposing the order 2 and 3 conditions in Table 4.4. The analysis of the corresponding families of three-stage explicit methods is left as exercise to the reader (see Exercise 8 at the end of this chapter). Famous examples of three-stage methods are the third order Kutta method

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

and Heun method

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}$$

having order 3.

The case $s = 4$ is also left to the reader (see Exercise 9 at the end of this chapter), but let us mention here two famous four-stage RK methods of order 4: the so-called *classical RK method*

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

and the *3/8-method*

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{3} & \frac{1}{3} & & & \\ \frac{2}{3} & -\frac{1}{3} & 1 & & \\ 1 & 1 & -1 & 1 & \\ \hline & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{array} \tag{4.21}$$

which is implemented in Program 4.1.

As above mentioned, relevant examples of explicit RK methods have been introduced by Karl Heun, whose brief portrait (based on https://mathshistory.st-andrews.ac.uk/Biographies/Heun/, [298]) is given below.

A Portrait of Karl Heun

Karl Heun was born in Wiesbaden (Germany) in 1859. After beginning his studies in mathematics and philosophy in 1878 in Göttingen, he moved to Halle in 1880 to study with Eduard Heine (become famous after the

(continued)

publication of his book on spherical harmonics in 1861), who died in 1881. He returned to Göttingen and started with his thesis, inspired by Heine, under the supervision of the astronomer Ernst Schering. The thesis was entitled *"Die Kugelfunctionen und Laméschen Functionen als Determinanten"* (The spherical harmonics and Lamé functions as determinants). After getting his doctorate he held a position as instructor at an agricultural winter school in Wehlau and got the qualification as teacher for secondary schools in Prussia.

Heun then worked as instructor in Uppingham (England) from 1883 to 1885. He complemented his studies in London and discussed his habilitation thesis in Munich in 1886; the thesis was entitled *"Uber lineare Differentialgleichungen zweiter Ordnung, deren Lösungen durch den Kettenbruchalgorithmus verknüpft sind"* (On linear second order differential equations whose solutions are linked by the continued fraction algorithm). He lectured in Munich from 1886 to 1889 but the absence of an adequate financial support led him leave Munich and move to Berlin, where we worked as a teacher from 1890 to 1902. In the meanwhile, he become quite famous in Germany, maybe due to a speech given at the Munich meeting of the Deutsche-Mathematiker-Vereinigung, later published as *"Die kinetischen Probleme der wissenschaftlichen Technik"* (The kinetic problems of scientific technology).

In 1902 Heun got his professorship for the vacant chair in technical mechanics at Technische Hochschule Karlsruhe, recommended by Felix Klein. He has never recovered from a bad stroke had in 1921; he retired in 1922 and remained in Karlsruhe until his death in 1929. His name is also associated to a differential equations, the Heun equation, which is a second order linear differential equation of the Fuchsian type with four singular points. This equation is a generalization of the Riemann hypergeometric differential equation (having three singular points) and plays a role mathematical physics, in the context of integrable systems.

Program 4.1 (3/8-Method)

```
% Function implementing the 3/8-method on a uniform grid,
% for the numerical solution of a d-dimensional ODE.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
% - y0: initial value;
% - h: constant stepsize.
```

(continued)

Program 4.1 (continued)

```
% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: d×N matrix whose i-th column y(:,i) stores the
%    approximate value in the i-th grid point, i=1,2,...,N.

function [t,y]=method38(problem,tspan,y0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0);
Id=eye(d);
y=zeros(d,N);
c=[0;1/3;2/3;1];
A=[0 0 0 0; 1/3 0 0 0; -1/3 1 0 0; 1 -1 1 0];
b=[1; 3; 3; 1]/8;
Y1=y0;
f1=f(problem,tspan(1)+c(1)*h,Y1);
Y2=y0+h*A(2,1)*f1;
f2=f(problem,tspan(1)+c(2)*h,Y2);
Y3=y0+h*(A(3,1)*f1+A(3,2)*f2);
f3=f(problem,tspan(1)+c(3)*h,Y3);
Y4=y0+h*(A(4,1)*f1+A(4,2)*f2+A(4,3)*f3);
f4=f(problem,tspan(1)+c(4)*h,Y4);
y(:,1)=y0+h*kron(b',Id)*[f1; f2; f3; f4];
for n=2:N
    Y1=y(:,n-1);
    f1=f(problem,t(n-1)+c(1)*h,Y1);
    Y2=y(:,n-1)+h*A(2,1)*f1;
    f2=f(problem,t(n-1)+c(2)*h,Y2);
    Y3=y(:,n-1)+h*kron(A(3,1:2),Id)*[f1;f2];
    f3=f(problem,t(n-1)+c(3)*h,Y3);
    Y4=y(:,n-1)+h*kron(A(4,1:3),Id)*[f1;f2;f3];
    f4=f(problem,t(n-1)+c(4)*h,Y4);
    y(:,n)=y(:,n-1)+h*kron(b',Id)*[f1; f2; f3; f4];
end
```

Example 4.4 Consider van der Pol oscillator (3.24) for $t \in [0, 10]$ and initial value $y_0 = [2 \ \ -2/3]^T$. Let us numerically solve this problem by using the 3/8-method (4.21). Table 4.5 shows the numerical results obtained for selected values of the stepsize. As visible from the numerical evidence, both the convergence and the order of convergence of the method are confirmed. The error reported in the table is computed as

$$\|\text{err}_{38\text{RK}}\|_\infty = \|y_{38\text{RK}} - y_{\text{ODE45}}\|_\infty,$$

(continued)

Example 4.4 (continued)
where $y_{38RK} \approx y(10)$ is computed by (4.21) and y_{ODE45} is the solution in $t = 10$ computed by the Matlab built-in function `ode45`, with high accuracy, given by

$$y_{\text{ODE45}} = [-1.914027891764918 \quad 0.446099168376746]^{\intercal}.$$

The table also shows again the results obtained by applying the second order Adams-Bashforth method (3.3), i.e.,

$$\|\text{err}_{\text{AB}}\|_\infty = \|y_{\text{AB}} - y_{\text{ODE45}}\|_\infty,$$

where $y_{AB} \approx y(10)$ is computed by (3.3). As expected, by using the same value of the stepsize, the 3/8-method (4.21) is able to provide higher accuracy, since it has higher order than that of (3.3). The number of vector field evaluations requested by both methods is listed in Table 4.5, for each chosen value of the stepsize: comparing the two methods in terms of accuracy, 3/8-method requires a lower number of function evaluations to reach a similar value of the error (396 evaluations of (4.21) needed to reach an error equal to $9.96 \cdot 10^{-5}$, vs 1601 of (3.3) requested to achieve an accuracy equal to $1.36 \cdot 10^{-4}$).

4.4 Fully Implicit Methods

Let us now turn our attention to fully implicit Runge-Kutta methods, whose seminal study has been given by Butcher in [57, 58]. The tableau of fully implicit methods

Table 4.5 Example 4.4: error in the final integration point associated to the application of the 3/8-method (4.21) and the Adams-Bashforth method (3.4) to the van der Pol problem (3.24). The estimation of the orders of both methods ($p_{38\text{RK}}$ and p_{AB}) is reported, computed as suggested by Equation (3.23), together with the requested number of vector field evaluations ($\text{fe}_{38\text{RK}}$ and fe_{AB})

h	$\|\text{err}_{38\text{RK}}\|_\infty$	$p_{38\text{RK}}$	$\text{fe}_{38\text{RK}}$	$\|\text{err}_{\text{AB}}\|_\infty$	p_{AB}	fe_{AB}
$h_0 = 0.1$	$9.96 \cdot 10^{-5}$		396	$3.10 \cdot 10^{-2}$		101
$h_0/2$	$5.91 \cdot 10^{-6}$	4.07	796	$8.23 \cdot 10^{-3}$	1.91	201
$h_0/4$	$3.62 \cdot 10^{-7}$	4.03	1596	$2.13 \cdot 10^{-3}$	1.95	401
$h_0/8$	$2.24 \cdot 10^{-8}$	4.01	3196	$5.41 \cdot 10^{-4}$	1.97	801
$h_0/16$	$1.39 \cdot 10^{-9}$	4.01	6396	$1.36 \cdot 10^{-4}$	1.99	1601

is given by

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \dots & a_{1s} \\ c_2 & a_{21} & a_{22} & \dots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \dots & a_{ss} \\ \hline & b_1 & b_2 & \dots & b_s \end{array} \tag{4.22}$$

where we recognize that A is a full matrix. We aim to present relevant examples of implicit methods, which depend on the choice of the quadrature points characterizing each method. Indeed, we have seen that RK methods (4.8) depend on their underlying quadrature formulae (4.3) and (4.6). The choice of such quadrature formulae (and, in particular, of their nodes) determines the corresponding RK method. In this section we present some classes of RK methods belonging to Gaussian, Radau and Lobatto quadrature formulae. We notice that the maximum order of a quadrature formula depending on s nodes is $2s$ and is achieved by Gaussian quadrature formulae (see, for instance, [170, 292]).

4.4.1 Gauss Methods

The first family of methods is well-known as family of *Gauss methods* or *Gauss-Legendre methods*. They are methods of maximal order $2s$, being s the number of internal stages. The nodes here considered are the zeros of Legendre orthogonal polynomials [332]. The one-stage method of order 2 is given by the following Butcher tableau

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} \tag{4.23}$$

and corresponds to the famous *midpoint method*

$$y_{n+1} = y_n + hf\left(t_n + \frac{1}{2}h, \frac{1}{2}\left(y_n + y_{n+1}\right)\right). \tag{4.24}$$

The method with $s = 2$ and order 4 is characterized by the Butcher tableau

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \tag{4.25}$$

while the method

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

is the three-stage formula of order 6. Higher order methods are listed, for instance, in [67].

4.4.2 Radau Methods

Radau methods are fully implicit methods (4.22) of order $2s - 1$, introduced in [58], but relevant seminal contributions have also been given in [87, 156]. These methods are based on Radau quadrature formulae, introduced by Radau in [295], based on the zeros of Jacobi orthogonal polynomials [332]. Radau points are generally classified in two classes: Radau IA points, if $c_1 = 0$, and Radau IIA points, if $c_s = 1$. Let us first consider Radau IA methods. The method of order 1, with $s = 1$, is characterized by the Butcher tableau

$$\begin{array}{c|c} 0 & 1 \\ \hline & 1 \end{array}$$

not fulfilling the row-sum condition (4.10). Oliver [281] observed that such a condition, at least for low order methods, is not mandatory: indeed, it is a simplifying assumption for the solution of order conditions, which is particularly useful in the derivation of high order methods.

The two-stage method of order 3 is given by the tableau

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array} \tag{4.26}$$

while the one with 3 internal stages is characterized by

$$\begin{array}{c|ccc} 0 & \frac{1}{9} & -\frac{1}{18}-\frac{\sqrt{6}}{18} & -\frac{1}{18}+\frac{\sqrt{6}}{18} \\ \frac{3}{5}-\frac{\sqrt{6}}{10} & \frac{1}{9} & \frac{11}{45}+\frac{7\sqrt{6}}{360} & \frac{11}{45}-\frac{43\sqrt{6}}{360} \\ \frac{3}{5}+\frac{\sqrt{6}}{10} & \frac{1}{9} & \frac{11}{45}+\frac{43\sqrt{6}}{360} & \frac{11}{45}-\frac{7\sqrt{6}}{360} \\ \hline & \frac{1}{9} & \frac{4}{9}+\frac{\sqrt{6}}{36} & \frac{4}{9}-\frac{\sqrt{6}}{36} \end{array}$$

and has order 5.

We finally consider Radau IIA methods. The one-stage method of order 1 corresponds to the Butcher tableau

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

that is the implicit Euler method (2.32). The two-stage method of order 3 is characterized by the tableau

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

while that depending on 3 internal stages is given by

$$\begin{array}{c|ccc} \frac{2}{5}-\frac{\sqrt{6}}{10} & \frac{11}{45}-\frac{7\sqrt{6}}{360} & \frac{37}{225}-\frac{169\sqrt{6}}{1800} & -\frac{2}{225}+\frac{\sqrt{6}}{75} \\ \frac{2}{5}+\frac{\sqrt{6}}{10} & \frac{37}{225}+\frac{169\sqrt{6}}{1800} & \frac{11}{45}+\frac{7\sqrt{6}}{360} & -\frac{2}{225}-\frac{\sqrt{6}}{75} \\ 1 & \frac{4}{9}-\frac{\sqrt{6}}{36} & \frac{4}{9}+\frac{\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{4}{9}-\frac{\sqrt{6}}{36} & \frac{4}{9}+\frac{\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

and has order 5. Last Radau method is at the basis of a famous code by E. Hairer and G. Wanner, named RADAU5, which is described in [195] (https://www.unige.ch/~hairer/software.html). The code is developed for problems of the form $My' = f(t, y)$ with possibly singular matrix M. An extension for delay differential equations, RADAR5, has been designed by Guglielmi and Hairer (https://www.unige.ch/~hairer/software.html).

4.4.3 Lobatto Methods

We finally present Lobatto methods, which are fully implicit methods (4.22) of order $2s-2$, introduced in [87, 156]. Such methods are based on Lobatto quadrature formulae, introduced by Lobatto in [254], based on the zeros of some Jacobi orthogonal polynomials [332], but different from those characterizing Radau quadrature formulae. Indeed, Lobatto quadrature points are always characterized by both $c_1 = 0$ and $c_s = 1$. Lobatto methods are divided into three classes: Lobatto IIIA methods, where the matrix A has always a row of zeros; Lobatto IIIB, where the matrix A has a column of zeros; Lobatto IIIC, where the matrix A has non-zero entries. Let us first consider Lobatto IIIA methods. The method of order 2, with $s = 2$, is given by

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

while that depending on 3 internal stages is characterized by the tableau

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

and has order 4.

Let us now present Lobatto IIIB methods, starting with the case $s = 2$ and order 2

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

not satisfying the row-sum condition (4.10). The method with $s = 3$ is characterized by the Butcher tableau

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{6} & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} & 0 \\ 1 & \frac{1}{6} & \frac{5}{6} & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

and order 4. We observe that, since the last column of Lobatto IIIB methods is the zero vector, they are all explicit in the computation of the last internal stage Y_s.

We finally present two Lobatto IIIC methods: the one of order 2, with $s = 2$,

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

and that with $s = 3$

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{3} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{6} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

of order 4.

4.5 Collocation Methods

So far we have analyzed numerical schemes providing the approximate solution of (1.1) in a prescribed number of selected points, according to the chosen stepsize. However, it is also possible to introduce numerical methods that provide a *dense output*, i.e., a continuous functional approximation to the solution of (1.1) obeying the so-called *collocation* principle. A collocation method computes the approximant, known as *collocation function*, within a finite dimensional space, denoted as *collocation space*; the technique to select a proper element of this space is normally based on interpolation in the chosen grid points plus the additional assumption that the approximant satisfies Eq. (1.1) at some selected points of the integration interval, denoted as *collocation points*. The choice of the collocation space generally relies

on algebraic polynomials, unless (1.1) describe a phenomenon whose qualitative behavior is not well described by algebraic polynomials and can be predicted in advance: in this case, one can use collocation spaces spanned by the most suitable functions for the description of the phenomenon (e.g., exponential functions, if (1.1) models phenomena with exponential decay; trigonometric functions, if (1.1) describes the dynamics of periodic phenomena). The presentation only deals with the case of algebraic polynomials, but the interested reader can refer to [227, 283] and references therein for the non-polynomial case.

We aim to construct a piecewise algebraic polynomial as collocation function. In particular, supposing to advance from t_n to t_{n+1}, we compute a unique algebraic polynomial $P_n(t)$ such that

$$\begin{aligned} P_n(t_n) &= y_n, \\ P_n'(t_n + c_i h) &= f(t_n + c_i h, P_n(t_n + c_i h)), \quad i = 1, 2, \ldots, s. \end{aligned} \tag{4.27}$$

In other terms, we require that $P_n(t)$ interpolates (t_n, y_n) and satisfies Equation (1.1) at the s internal points $t_n + c_i h$, for $i = 1, 2, \ldots, s$. Once $P_n(t)$ is computed from (4.27), the numerical solution y_{n+1} is then given by

$$y_{n+1} = P_n(t_{n+1}). \tag{4.28}$$

The set of $s+1$ constraints (4.27) can be recast in a convenient matrix form. Indeed, suppose that $P_n(t)$ is linear combination of $s+1$ algebraic polynomials

$$\{\varphi(\eta),\ \psi_1(\eta),\ \psi_2(\eta),\ \ldots,\ \psi_s(\eta),\ \eta \in [0, 1]\} \tag{4.29}$$

in the following Runge-Kutta-like form

$$P_n(t_n + \eta h) = \varphi(\eta) y_n + h \sum_{i=1}^{s} \psi_i(\eta) f\left(t_n + c_i h, P_n(t_n + c_i h)\right). \tag{4.30}$$

Conditions (4.27), applied to (4.30), are equivalent to

$$\begin{aligned} \varphi(0) &= 1, \quad \varphi'(c_i) = 0, \\ \psi_i(0) &= 0, \quad \psi_i'(c_j) = \delta_{ij}, \end{aligned} \tag{4.31}$$

for $i, j = 1, 2, \ldots, s$, being δ_{ij} the Kronecker delta. As a consequence, each basis function in (4.29) satisfies $s+1$ constraints, therefore they are algebraic polynomials of degree at most s. Let us assume the following form for each of them

$$\begin{aligned} \varphi(\eta) &= \alpha_0 + \alpha_1 \eta + \ldots + \alpha_s \eta^s, \\ \psi_i(\eta) &= \beta_{i0} + \beta_{i1} \eta + \ldots + \beta_{is} \eta^s, \quad i = 1, 2, \ldots, s. \end{aligned}$$

Conditions (4.31) on $\varphi(\eta)$ are equivalent to $\alpha_0 = 1$ and $\alpha_1, \alpha_2, \ldots, \alpha_s$ satisfying the linear system $C\alpha = 0$, with

$$C = \begin{bmatrix} 1 & 2c_1 & \ldots & sc_1^{s-1} \\ 1 & 2c_2 & \ldots & sc_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 2c_s & \ldots & sc_s^{s-1} \end{bmatrix}, \quad \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_s \end{bmatrix}.$$

We observe that $\det(C) = s!\det(V(c_1, c_2, \ldots, c_s))$, being $V(c_1, c_2, \ldots, c_s)$ the Vandermonde matrix on the vector $[c_1, c_2, \ldots, c_s]$

$$V(c_1, c_2, \ldots, c_s) = \begin{bmatrix} 1 & c_1 & \ldots & c_1^{s-1} \\ 1 & c_2 & \ldots & c_2^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \ldots & c_s^{s-1} \end{bmatrix}.$$

It is well-known (see, for instance, [292, 329]) that a Vandermonde matrix on distinct nodes is non-singular, therefore also C is non-singular. Hence the homogeneous system $C\boldsymbol{\alpha} = 0$ only admits the trivial solution $\boldsymbol{\alpha} = 0$, so $\varphi(\eta) = 1$. Similarly, we can prove that each linear system for the computation of the coefficients of $\psi_i(\eta)$, $i = 1, 2, \ldots, s$, admits a unique solution. The proof is left to the reader.

Hence, there exists a unique algebraic polynomial $P_n(t)$ of the form (4.30) satisfying (4.27), with $\varphi(\eta) = 1$, assuming that $c_i \neq c_j$, $i \neq j$.

Guillou and Soulé in [185] as well as Wright in [347] independently proved that collocation methods (4.28) are implicit Runge-Kutta methods, as proved by the following theorem.

Theorem 4.6 (Guillou and Soulé; Wright) *A collocation method (4.28) is equivalent to a s-stage RK method (4.8), with*

$$a_{ij} = \int_0^{c_i} L_j(u)du, \quad b_i = \int_0^1 L_i(u)du, \tag{4.32}$$

for $i, j = 1, 2, \ldots, s$, where $L_i(u)$ is the i-th fundamental Lagrange polynomial

$$L_i(u) = \prod_{\substack{j=1 \\ j \neq i}}^{s} \frac{u - c_j}{c_i - c_j}. \tag{4.33}$$

Proof According to conditions (4.27), $P_n'(t)$ is the interpolation polynomial of degree $s-1$ on the nodes

$$(t_n + c_i h,\, P_n'(t_n + c_i h)), \quad i = 1, 2, \ldots, s.$$

Its Lagrangian formulation is then given by

$$P_n'(t_n + \eta h) = \sum_{i=1}^{s} L_i(\eta) P_n'(t_n + c_i h), \tag{4.34}$$

being $L_i(\eta)$ the i-th fundamental Lagrange polynomial (4.33). Side-by-side integration from 0 to c_j leads to

$$\begin{aligned} P_n(t_n + c_j h) - y_n &= h \sum_{i=1}^{s} \left(\int_0^{c_j} L_j(u)\mathrm{d}u \right) P_n'(t_n + c_j h) \\ &= h \sum_{i=1}^{s} a_{ij} f(t_n + c_j h,\, P(t_n + c_j h)), \end{aligned}$$

for $j = 1, 2, \ldots, s$, while integrating from 0 to 1 gives

$$\begin{aligned} y_{n+1} - y_n &= h \sum_{i=1}^{s} \left(\int_0^{1} L_i(u)\mathrm{d}u \right) P_n'(t_n + c_i h) \\ &= h \sum_{i=1}^{s} b_i f(t_n + c_i h,\, P(t_n + c_i h)). \end{aligned}$$

Denoting $Y_j = P_n(t_n + c_j h)$ gives the thesis. □

The result provided by Theorem 4.6 suggests us how to construct collocation based Runge-Kutta methods. Clearly, not all implicit Runge-Kutta methods are collocation methods: a useful characterization is given by the following theorem.

Theorem 4.7 *A s-stage RK method (4.8) of order at least s and depending on distinct nodes is a collocation method, i.e., its coefficients satisfy (4.32), if and only if*

$$\sum_{j=1}^{s} a_{ij} c_j^{k-1} = \frac{c_i^k}{k}, \quad i, k = 1, 2, \ldots, s. \tag{4.35}$$

Proof Suppose that $\pi(u)$ is a monomial of degree at most $s-1$. Lagrange interpolation formula on the nodes $c_1, c_2, \ldots, c_s$ gives

$$\pi(u) = \sum_{j=1}^{s} L_j(u)\pi(c_j).$$

Integrating from 0 to c_i leads to

$$\int_0^{c_i} \pi(u)\mathrm{d}u = \sum_{j=1}^{s} \left(\int_0^{c_i} L_j(u)\mathrm{d}u \right) \pi(c_j), \quad i = 1, 2, \ldots, s,$$

or, equivalently,

$$\int_0^{c_i} \pi(u)\mathrm{d}u = \sum_{j=1}^{s} a_{ij}\pi(c_j), \quad i = 1, 2, \ldots, s,$$

that is (4.35). □

We observe that setting $k = 1$ in (4.35) leads to the row-sum condition (4.10). According to Theorem 4.7, all Gaussian, Radau IIA and Lobatto IIIA formulae are collocation methods. The check is left to the reader (see Exercise 10 at the end of this chapter).

As a further consequence, the maximum attainable order for a collocation method is $2s$, that is the maximum attainable order of a s-stage implicit Runge-Kutta method. The solution computed by a collocation method (4.28) inherits the order of the corresponding Runge-Kutta method in the grid points. However, a more general result on the *uniform* order of collocation-based Runge-Kutta method, i.e., the order observed in any point in the interval $[t_n, t_{n+1}]$, is proved in Chap. 7 and the consequences of this issue are also discussed.

Example 4.5 Consider the Runge-Kutta method based on two Gaussian points (4.25), which is known to be a collocation method. We aim to provide the expression of the corresponding collocation polynomial (4.30), of the type

$$P_n(t_n + \eta h) = y_n + h\left(\psi_1(\eta) f_1 + \psi_2(\eta) f_2\right),$$

(continued)

Example 4.5 (continued)
where $f_1 = (t_n + c_1 h, P_n(t_n + c_1 h))$ and $f_2 = (t_n + c_2 h, P_n(t_n + c_2 h))$. We impose the interpolation and collocation conditions (4.27), assuming that

$$\psi_1(\eta) = \beta_{10} + \beta_{11}\eta + \beta_{12}\eta^2,$$
$$\psi_2(\eta) = \beta_{20} + \beta_{21}\eta + \beta_{22}\eta^2.$$

The unknown coefficients of the above polynomials satisfy the linear system

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2c_1 & 0 & 0 & 0 \\ 0 & 1 & 2c_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2c_1 \\ 0 & 0 & 0 & 0 & 1 & 2c_2 \end{bmatrix} \begin{bmatrix} \beta_{10} \\ \beta_{11} \\ \beta_{11} \\ \beta_{20} \\ \beta_{21} \\ \beta_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

whose solution is given by

$$\begin{bmatrix} \beta_{10} \\ \beta_{11} \\ \beta_{12} \\ \beta_{20} \\ \beta_{21} \\ \beta_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1+\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{2} \\ 0 \\ \frac{1-\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} \end{bmatrix}$$

Hence, the collocation polynomial (4.30) associated to (4.25) is given by

$$P_n(t_n+\eta h) = y_n + h\left(\left(\frac{1+\sqrt{3}}{2}\eta - \frac{\sqrt{3}}{2}\eta^2\right) f_1 + \left(\frac{1-\sqrt{3}}{2}\eta + \frac{\sqrt{3}}{2}\eta^2\right) f_2\right).$$

The reader can easily check that the matrix A in (4.25) can be recovered as follows:

$$A = \begin{bmatrix} \psi_1(c_1) & \psi_2(c_1) \\ \psi_1(c_2) & \psi_2(c_2) \end{bmatrix}.$$

We have realized that only a subset of implicit Runge-Kutta formulae are collocation methods and certainly extending this subset to a larger class may provide all the benefits emerging from dense output methods. For this reason, the literature has provided a number of contributions in order to provide *continuous Runge-Kutta methods*. One of the first systematic collections of results on continuous Runge-Kutta methods has been provided in [22], where several techniques to provide continuous extensions keeping the same internal stages or adding new ones is presented and analyzed in details. In this framework, collocation methods appear to be a particular class of continuous RK methods (see [22] and references therein).

4.6 Exercises

1. Compute the two-stage method (4.20) with minimal error constant.
2. Prove that, for all explicit four-stage Runge-Kutta methods of maximal order, $c_4 = 1$.
3. Suppose that P_n defined in (4.30) is the collocation polynomial of a given Runge-Kutta method (4.8). Find the expression of the matrix A and the vector b of this method, in terms of the basis functions (4.29).
4. Write a software in the programming language you prefer that implements the two-stage RK methods on Gaussian nodes (4.25) by exploiting its collocation polynomial, computed in Example 4.5. In particular, the updated value y_{n+1}, given y_n, has to be computed by $y_{n+1} = P_n(t_{n+1})$.
5. Compute elementary differentials, elementary weights and order conditions associated to the following rooted trees

$t_1 =$  , $t_2 =$  , $t_3 =$  , $t_4 =$ 

   Is there any relationship among the elementary weights $\Phi(t_1)$, $\Phi(t_2)$, $\Phi(t_3)$, $\Phi(t_4)$?
6. A non-empty set G with an internal operation $\circ : G \times G \to G$ is a *group* if
   - $(a \circ b) \circ c = a \circ (b \circ c)$, for any $a, b, c \in G$;
   - there exists $z \in G$ such that $a \circ z = z \circ a = a$, for any $a \in G$;
   - for any $a \in G$, there exists $a' \in G$, such that $a \circ a' = a' \circ a = z$.

   Prove that the set of Runge-Kutta methods, seen as maps $y_0 \to y_1$ defined by the B-series (4.18), form a group with the usual composition of maps as internal operation (observe that the compositions of B-series is still a B-series). This is a famous structure, known in literature as Butcher group [67].
7. Compute the error constants of all explicit methods given in Sect. 4.3.
8. Analyze the family of explicit Runge-Kutta methods (4.8) depending on 3 internal stages. Provide examples of convergent methods, also of maximal order.

9. Analyze the family of explicit Runge-Kutta methods (4.8) depending on 4 internal stages. Provide examples of convergent methods, also of maximal order.
10. Using Theorem 4.7, check that all Gaussian, Radau IIA and Lobatto IIIA formulae are collocation methods.

Chapter 5
Multivalue Methods

Even though multistep and Runge-Kutta methods developed individually and separately, they have always had a common core. That is, they are each built up from two basic operations and nothing more: the evaluation of the function f and the calculation of linear combinations of existing vectors.

(John C. Butcher [68])

So far we have analyzed numerical methods for ODEs (1.1) providing an approximated value of the solution at each point of the grid (2.1). Such an approximation is computed according to a multistep principle as in (3.1), or by a multistage strategy as in (4.8). In both cases, the discretization only involves samples of the solution to (1.1) and linear combinations of vector field evaluations, while other solution related quantities do not play any role. As we have seen, for both LMMs and RK methods, order barriers do not permit the construction of methods of arbitrarily high orders; moreover, the order of a method depends on the number of steps for LMMs and on the number of internal stages for RK methods. However, involving many steps in LMMs may affect their stability properties (as we clarify in Chap. 6), while heightening the number of internal stages in RK methods has a direct influence on their computational cost.

In the recent history of numerical analysis for ODEs, a third kind of strategy for the improvement of the drawbacks of multistep and multistage techniques has been provided, mostly by J. Butcher, giving rise to the so-called family of *multivalue* numerical methods. The basic principles of a multivalue discretization are briefly provided here; the interested reader can find a fully detailed presentation in [64, 67, 132, 228].

5.1 Multivalue Numerical Dynamics

A multivalue numerical method for the solution of the initial value problem (1.1) provides a discrete dynamics as described in Fig. 5.1.

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_5

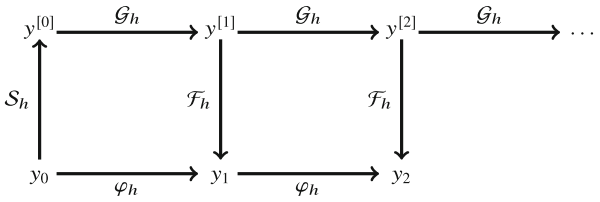


Fig. 5.1 Dynamics of a multivalue numerical method

As displayed in Fig. 5.1, multivalue numerical methods compute a vector of r values, denoted as $y^{[n+1]} \in \mathbb{R}^{rd}$, assuming that the analogous vector $y^{[n]}$ of approximations in the previous point is given. The updated vector of approximations is computed according to the map

$$\mathcal{G}_h : y^{[n]} \in \mathbb{R}^{rd} \to y^{[n+1]} \in \mathbb{R}^{rd},$$

denoted as *forward* procedure. The generic vector of approximations $y^{[n]}$ provides r quantities related to the solution of the problem; it does not only (and not necessarily) contain the approximation of the solution in the grid points (as it happens for LMMs (3.1) and RK methods (4.8)), but also solution related quantities, such as linear combination of the derivatives, evaluations of the vector field and so on. For instance, consider a system of ODEs (1.1) describing the motion of a system of particles for which an approximation of the solution and the velocity is required: in this case a multivalue method provides, at each step point, a vector approximating the solution of (1.1) and its first derivative. Moreover, involving more quantities in the discretized dynamics may allow the introduction of additional degrees of freedom in the method which can be exploited to improve the barriers of multistep and multistage methods. An example of vector $y^{[n]}$ is given, for instance, by the so-called *Nordsieck vector*

$$y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix} \approx \begin{bmatrix} y(t_n) \\ hy'(t_n) \\ \vdots \\ h^{r-1} y^{(r-1)}(t_n) \end{bmatrix}, \tag{5.1}$$

that provides an approximation of the first $r-1$ scaled derivatives of the solution to (1.1).

Multivalue methods are clearly not self-starting: indeed, a *starting* procedure $\mathcal{S}_h$, defined by

$$\mathcal{S}_h : y_0 \in \mathbb{R}^d \to y^{[0]} \in \mathbb{R}^{rd},$$

is required for the computation of the missing starting vector $y^{[0]}$.

Moreover, it is possible to recover the approximate solution at each step point by the projection map

$$\mathcal{F}_h : y^{[n]} \in \mathbb{R}^{rd} \to y_n \in \mathbb{R}^d,$$

denoted in the literature as *finishing* procedure $\mathcal{F}_h$.

It was proved in [192] that, for any given forward and finishing procedures, there exist a unique starting procedure and a unique one-step method

$$y_{n+1} = \varphi_h(y_n),$$

such that

$$\mathcal{G}_h \circ \mathcal{S}_h = \mathcal{S}_h \circ \varphi_h,$$

with $\mathcal{F}_h \circ \mathcal{S}_h$ equal to the identity map. Such a formal one-step map φ_h is called *underlying one-step method*. The analysis of the underlying one-step method is very relevant, since many of its properties are inherited by the forward procedure of the corresponding multivalue methods. We will see this aspect, for instance, in the analysis of symmetry and symplecticity of numerical methods provided in Chap. 8.

5.2 General Linear Methods Representation

A classical representation of multivalue methods is usually given in the form of General Linear Methods (GLMs) [67, 228]

$$\begin{aligned} Y_i &= h \sum_{j=1}^{s} a_{ij} f(t_n + c_j h, Y_j) + \sum_{j=1}^{r} u_{ij} y_j^{[n]}, \quad i = 1, 2, \ldots, s, \\ y_i^{[n+1]} &= h \sum_{j=1}^{s} b_{ij} f(t_n + c_j h, Y_j) + \sum_{j=1}^{r} v_{ij} y_j^{[n]}, \quad i = 1, 2, \ldots, r, \end{aligned} \tag{5.2}$$

where $c_1, c_2, \ldots, c_s$ are the values of the nodes, as in the case of RK methods (4.8). Assuming that $Y_i \approx y(t_n + c_i h)$, $i = 1, 2, \ldots, s$, GLMs combine a multistage strategy (the numerical dynamics described by (5.2) requires the computation of the internal stages Y_i, $i = 1, 2, \ldots, s$, at each step) with a multivalue strategy for the computation of the $y_i^{[n+1]}$, $i = 1, 2, \ldots, r$. As visible from their representation formulae (5.2), GLMs are uniquely defined by the coefficient matrices

$$A \in \mathbb{R}^{s \times s}, \quad U \in \mathbb{R}^{s \times r}, \quad B \in \mathbb{R}^{r \times s}, \quad V \in \mathbb{R}^{r \times r},$$

which can be conveniently collected in the Butcher tableau

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right]. \tag{5.3}$$

Such a representation in terms of a $(s + r) \times (s + r)$ partitioned matrix has been introduced for the first time by K. Burrage and J. Butcher in [50] and then extensively applied in the context of GLMs.

A GLM admits the following compact notation

$$\begin{cases} Y = h(A \otimes I)F + (U \otimes I)y^{[n-1]}, \\ y^{[n]} = h(B \otimes I)F + (V \otimes I)y^{[n-1]}, \end{cases} \tag{5.4}$$

where $\otimes$ denotes the usual Kronecker tensor product and $I \in \mathbb{R}^{d\times d}$ is the identity matrix and

$$y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix} \in \mathbb{R}^{rd}, \quad Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{bmatrix} \in \mathbb{R}^{rd}, \quad F = \begin{bmatrix} f(t_n + c_1 h, Y_1) \\ f(t_n + c_2 h, Y_2) \\ \vdots \\ f(t_n + c_s h, Y_s) \end{bmatrix} \in \mathbb{R}^{sd}.$$

One can appreciate that GLMs depend on $(s + r)^2 + s$ coefficients, while RK methods depend on $s^2 + 2s$ coefficients (the entries of the matrix A and the vectors b and c). As previously observed, the dependence on a larger number coefficients (which is certainly the case, in comparison with RK methods, if $r > 1$) can be exploited to break the order barriers affecting RK methods and obtain more accurate methods without increasing the computational cost that depends, as in the RK case, on the coefficient matrix A. Again, if A is strictly lower triangular, the GLM is explicit; if A is a full matrix, the method is implicit.

Finally, few historical notes. The name *generalized multistep methods* has been used for the first time by Gragg and Stetter [178] in 1964. Further contributions in the development of a theory of multivalue-multistage integration methods have been provided by Butcher from 1965 on (see [67] and references therein), Gear [172], Dahlquist [112], Donelson and Hansen [155], Jackiewicz and Tracogna [229]. The monograph [228] authored by Zdzislaw Jackiewicz (Swiebodzin, 1950) is a comprehensive presentation totally devoted to GLMs.

Example 5.1 All numerical methods studied in the previous chapters can be regarded as GLMs. Indeed, linear multistep methods (3.1) with $\alpha_k = 1$ are GLMs (5.2) with $r = 2k$, $s = 1$ and Butcher tableau (5.3) given by

$$\left[\begin{array}{c|cccccccc} \beta_0 & -\alpha_0 & \cdots & -\alpha_{k-2} & -\alpha_{k-1} & \beta_1 & \cdots & \beta_{k-1} & \beta_k \\ \hline \beta_0 & -\alpha_0 & \cdots & -\alpha_{k-2} & -\alpha_{k-1} & \beta_1 & \cdots & \beta_{k-1} & \beta_k \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 \end{array}\right].$$

The vector computed at each step is then given by

$$y^{[n]} = \begin{bmatrix} y_n \\ y_{n+1} \\ \vdots \\ y_{n+k-1} \\ f(t_n, y_n) \\ f(t_{n+1}, y_{n+1}) \\ \vdots \\ f(t_{n+k-1}, y_{n+k-1}) \end{bmatrix}.$$

Runge-Kutta methods (4.8) are GLMs with $r = 1$, Butcher tableau (5.3) given by

$$\left[\begin{array}{ccc|c} a_{11} & \cdots & a_{1s} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{s1} & \ldots & a_{ss} & 1 \\ \hline b_1 & \cdots & b_s & 1 \end{array}\right]$$

and

$$y^{[n]} = \begin{bmatrix} y_n \end{bmatrix}.$$

5.3 Convergence Analysis

A theory of multivalue methods is not only useful to develop new methods with better accuracy, but also to create a unifying approach to analyze the properties of a numerical method for ODEs, e.g., convergence, consistency and stability. Here we present a unifying convergence analysis, based on the representation of multivalue methods as GLMs (5.2). As a consequence, once a numerical method for (1.1) is represented as GLM, it automatically inherits the theoretical results here presented.

Let us apply the GLM (5.4) to the problem $y'(t) = 1$, obtaining

$$\begin{cases} Y = hAe + Uy^{[n]}, \\ y^{[n+1]} = hBe + Vy^{[n]}, \end{cases} \tag{5.5}$$

being $e = [1\ 1\ \cdots\ 1]^{\intercal} \in \mathbb{R}^s$. Assume the existence of two vectors $\alpha, \beta \in \mathbb{R}^r$ such that

$$y^{[n]} = \alpha y(t_n) + \beta h y'(y_n) + O(h^2) \tag{5.6}$$

and, moreover,

$$Y_i = y(t_n + c_i h) + O(h), \quad i = 1, 2, \ldots s. \tag{5.7}$$

Replacing (5.6) and (5.7) into (5.5) leads to

$$\begin{cases} y(t_n)e + chy'(t_n) = hAe + U(\alpha y(t_n) + \beta h y'(t_n)) + O(h^2), \\ \alpha y(t_n) + \alpha h y'(t_n) + \beta h y'(t_n) = hBe + V(\alpha y(t_n) + \beta h y'(t_n)) + O(h^2). \end{cases}$$

Comparing the $O(1)$ and $O(h)$ terms leads to the following definition.

Definition 5.1 A GLM (5.2) is *consistent* if there exist two vectors $\alpha, \beta \in \mathbb{R}^r$ such that

$$U\alpha = e, \quad V\alpha = \alpha, \quad Be + V\beta = \alpha + \beta \tag{5.8}$$

and it is *stage-consistent* if

$$Ae + U\beta = c. \tag{5.9}$$

Let us now apply the GLM (5.4) to the problem $y'(t) = 0$, obtaining

$$y^{[n+1]} = Vy^{[n]} = V^{n+1}y^{[0]}.$$

As a consequence, a stable behavior of the numerical solution provided by (5.4) requires the boundedness of the powers V^n, for any $n \geq 0$, motivating the following definition.

Definition 5.2 A GLM (5.4) is *zero-stable* if there exists a constant $C > 0$ such that

$$\|V^n\| \leq C, \quad n \geq 0.$$

As seen for linear multistep methods, a more practical way to analyze zero-stability is the verification of the so-called root condition. Such a condition, in the case of GLMs, has to be fulfilled by the minimal polynomial of the matrix V, as stated by the following result (compare [67, 228]).

Theorem 5.1 *A GLM (5.4) is zero-stable if each root of the minimal polynomial of the coefficient matrix V has modulus strictly less than 1 or it has modulus one but it is simple.*

Let us now provide a definition of convergence for GLMs.

Definition 5.3 A GLM (5.2) is *convergent* if there exists a nonzero vector $\alpha \in \mathbb{R}^r$ and a starting procedure $\mathcal{S}_h : \mathbb{R}^d \to \mathbb{R}^{rd}$ satisfying

$$\lim_{h \to 0} (\mathcal{S}_h(y_0))_i = \alpha_i y_0, \quad i = 1, 2, \ldots, r,$$

such that for any $\bar{t} > t_0$, the sequence of vectors $\{y^{[n]}\}_{n \in \mathbb{N}}$, computed by using n steps of (5.2) with stepsize $h = (\bar{t} - t_0)/n$ and starting value $y^{[0]} = \mathcal{S}_h(y_0)$, converges to $\alpha y(\bar{t})$.

A Lax equivalence theorem can be provided also for GLMs. The proof is here omitted, but the interested reader can find it in [67, 228].

Theorem 5.2 *A GLM (5.2) is convergent if and only if it is consistent and zero-stable.*

We conclude this section with a definition of order for GLMs. Let us assume that $y_i^{[n]}$, $i = 1, 2, \ldots, r$, is an approximation of order p of a linear combination of the solution to (1.1) and its derivatives in the point t_n, i.e.,

$$y_i^{[n]} = \sum_{k=0}^{p} q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \ldots, r, \tag{5.10}$$

where $q_{ik} \in \mathbb{R}$ are the scalars of combination, with indices $i = 1, 2, \ldots, r$ and $k = 0, 1, \ldots, p$. The integer p is the *order* of the GLM (5.2). Suppose that the internal stages satisfy

$$Y_i^{[n]} = y(t_{n-1} + c_i h) + O(h^{q+1}), \quad i = 1, 2, \ldots, r, \tag{5.11}$$

i.e., they are approximations of order q to the solution of (1.1) in the points $t_{n-1} + c_i h$, $i = 1, 2, \ldots, s$. In this case, we say that the GLM (5.2) has *stage-order* q.

The existing literature has provided two ways to analyze the order of multivalue methods: an extension of rooted trees and B-series theory, developed by Butcher [67, 68] and an extension of Albrecht theory [4–6], mostly relying on Taylor series arguments, provided by Jackiewicz and coauthors [78, 228].

Let us provide here only the set of conditions ensuring high stage-order methods, i.e., methods of order p and stage-order $q = p$. We collect the parameters q_{ik} appearing in (5.10) and (5.11) in the vectors q_k, $k = 0, 1, \ldots, p$, defined by

$$q_k = \left[q_{1k}\ q_{2k}\ \ldots\ q_{rk} \right]^{\mathsf{T}} \in \mathbb{R}^r, \quad k = 0, 1, \ldots, p.$$

Then, a GLM (5.2) has order p and stage-order $q = p$ if and only if

$$\frac{c^k}{k!} - \frac{A c^{k-1}}{(k-1)!} - U q_k = 0, \quad k = 1, 2, \ldots, p, \tag{5.12}$$

and

$$\sum_{l=0}^{k} \frac{q_{k-l}}{l!} - \frac{B c^{k-1}}{(k-1)!} - V q_k = 0, \quad k = 1, 2, \ldots, p. \tag{5.13}$$

A detailed proof is given in [67, 228]. We observe that above order and stage-order conditions can be applied also when $q = p - 1$, as highlighted in [228]. General

order conditions when $q \neq p-1$, p require to employ the general Butcher order theory, described in [67, 68]. We also notice that, in correspondence of $k = 1$, consistency (5.8) and stage-consistency (5.9) can be recovered, with $\alpha = q_0$ and $\beta = q_1$.

Example 5.2 We provide an example of explicit one-stage GLM (5.4) with $r = 2$, depending on the Butcher tableau

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} 0 & u_1 & u_1 \\ \hline b_1 & v_{11} & v_{12} \\ b_2 & v_{21} & v_{22} \end{array}\right].$$

assuming that the vector of approximations

$$y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \end{bmatrix}$$

gives $y_1^{[n]} = y_n$. Correspondingly, α and β in (5.8) and (5.9) are

$$\alpha = \begin{bmatrix} 1 \\ \alpha_2 \end{bmatrix}, \quad \beta = \begin{bmatrix} 0 \\ \beta_2 \end{bmatrix},$$

with $\alpha_2, \beta_2 \in \mathbb{R}$. Consistency and stage-consistency conditions yield

$$u_1 = 1 - \frac{\alpha_2 c}{\beta_2}, \quad \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = \begin{bmatrix} 1 - v_{12}\alpha_2 \\ (1 - v_{22})\alpha_2 \end{bmatrix},$$

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 - v_{12}\beta_2 \\ \alpha_2 + (1 - v_{22})\beta_2 \end{bmatrix}, \quad \begin{bmatrix} u_{21} \\ u_{22} \end{bmatrix}, \quad u_2 = \frac{c}{\beta_2}.$$

The resulting Butcher tableau

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} 0 & 1 - \frac{\alpha_2 c}{\beta_2} & \frac{c}{\beta_2} \\ \hline 1 - v_{12}\beta_2 & 1 - v_{12}\alpha_2 & v_{12} \\ \alpha_2 + (1 - v_{22})\beta_2 & (1 - v_{22})\alpha_2 & v_{22} \end{array}\right]$$

(continued)

Example 5.2 (continued)
depends on 5 degrees of freedom that are now used to solve order and stage order conditions up to a certain order p. We observe that order and stage-order conditions (5.13) and (5.12) depend on the vectors

$$q_k = \begin{bmatrix} 0 \\ \\ q_{k2} \end{bmatrix}, \quad q_{k2} \in \mathbb{R}.$$

Since

$$Uq_k = \frac{c^k}{k!},$$

we obtain

$$q_{k2} = \frac{\beta_2 c^{k-1}}{k!}.$$

As a consequence, in order to analyze the conditions of order 2, we assume

$$q_2 = \begin{bmatrix} 0 \\ \\ \dfrac{\beta_2 c}{2} \end{bmatrix}$$

and impose the conditions of order 2, i.e.,

$$q_2 + \beta + \frac{\alpha}{2} - Bc - Vq_2 = 0,$$

obtaining

$$\begin{bmatrix} v_{12} \\ \\ v_{22} \end{bmatrix} = \begin{bmatrix} \dfrac{2c-1}{\beta_2 c} \\ \\ \dfrac{\alpha_2(2c-1) + \beta_2(c-2)}{\beta_2 c} \end{bmatrix}.$$

We observe that the eigenvalues of the corresponding matrix V are 1 and $\frac{c-2}{c}$. Hence, zero-stability occurs if $c > 1$ and one can prove that, in this example, it is not compatible with order 3. As a consequence, we have developed a family of order and stage-order 2 methods depending on 3 free parameters (c, α_2 and β_2) which can be chosen, for instance, to have good stability properties according to the notions provided in Chap. 6.

(continued)

Example 5.2 (continued)

If we choose $c = \frac{3}{2}, \alpha_2 = \beta_2 = 1$, we obtain the method (5.4) with Butcher tableau

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} 0 & -\frac{1}{2} & \frac{3}{2} \\ \hline -\frac{1}{3} & -\frac{1}{3} & \frac{4}{3} \\ 1 & 0 & 1 \end{array}\right], \tag{5.14}$$

and

$$y^{[n]} \approx \begin{bmatrix} y(t_n) \\ y(t_n) + hy'(t_n) + \frac{3}{4}h^2 y''(t_n) \end{bmatrix}.$$

The starting vector is then given by

$$y^{[0]} = \begin{bmatrix} y_0 \\ y_0 + hf(t_0, y_0) + \frac{3}{4}h^2 f_y(t_0, y_0) f(t_0, y_0) \end{bmatrix}.$$

This method, denoted as GLM2, requires a single function evaluation at each step, as it happens for the explicit Euler method (2.19), but its order of convergence is twice as that of Euler method. We provide an implementation of GLM2 in Program 5.1.

Program 5.1 (GLM2 Method)

```
% Function implementing GLM2 method (5.14) on a uniform grid,
% for the numerical solution of a d-dimensional ODE.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
% - y0: initial value;
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: d×N matrix whose i-th column y(:,i) stores the
%      approximate value in the i-th grid point, i=1,2,...,N.
```

(continued)

Program 5.1 (continued)

```
function [t,y]=GLM2(problem,tspan,y0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0);
y=zeros(2*d,N);
Id=eye(d);
% Starting procedure, requiring the Jacobian fy of f
start=[y0; y0+h*f(problem,tspan(1),y0)+...
    3*h^2*fy(problem,tspan(1),y0)*f(problem,tspan(1),y0)/4];
U=[-1/2 3/2];
B=[-1/3; 1];
V=[-1/3 4/3; 0 1];
c=3/2;
Y=kron(U,Id)*start;
y(:,1)=h*kron(B,id)*f(problem,tspan(1)+c*h,Y)
        +kron(V,Id)*start;
for i=2:N
    Y=kron(U,Id)*y(:,i-1);
    y(:,i)=h*kron(B,id)*f(problem,t(i-1)+c*h,Y)...
          +kron(V,Id)*y(:,i-1);
end
```

Example 5.3 We aim to solve van der Pol problem (3.24), with $t \in [0, 10]$ and initial value $y_0 = [2 \ \ -2/3]^\intercal$. We provide the comparison of the performances achieved by the explicit Euler method (2.19) and the GLM2 method (5.14). We compute the errors

$$\|\text{err}_{\text{EUL}}\|_\infty = \|y_{\text{EUL}} - y_{\text{ODE45}}\|_\infty, \quad \|\text{err}_{\text{GLM2}}\|_\infty = \|y_{\text{GLM2}} - y_{\text{ODE45}}\|_\infty,$$

where $y_{\text{EUL}} \approx y(10)$ is computed by (2.19), $y_{\text{GLM2}} \approx y(10)$ is computed by (5.14) and y_{ODE45} is the solution in $t = 10$ computed by the Matlab built-in function `ode45`, as in Example 4.4. As shown in Table 5.1, convergence and orders of both methods are confirmed by the numerical evidence. Moreover, by employing almost the same number of function evaluations, GLM2 method is more accurate and efficient than the explicit Euler method.

Table 5.1 Example 3.10: error in the final integration point associated to the application of the explicit Euler (2.19) and the GLM2 (5.14) methods to (3.24). The estimation of the orders of both methods (p_{EUL} and p_{GLM2}) is reported, computed as suggested by Eq. (3.23), together with the requested number of vector field evaluations (fe_{EUL} and fe_{GLM2})

h	$\|\text{err}_{\text{EUL}}\|_\infty$	p_{EUL}	fe_{EUL}	$\|\text{err}_{\text{GLM2}}\|_\infty$	p_{GLM2}	fe_{GLM2}
$h_0 = 0.1$	1.57		100	$2.49 \cdot 10^{-2}$		102
$h_0/2$	$2.67 \cdot 10^{-1}$	2.55	200	$6.25 \cdot 10^{-3}$	1.99	202
$h_0/4$	$1.01 \cdot 10^{-1}$	1.40	400	$1.58 \cdot 10^{-3}$	1.98	402
$h_0/8$	$5.14 \cdot 10^{-2}$	0.97	800	$3.97 \cdot 10^{-4}$	1.99	802
$h_0/16$	$2.58 \cdot 10^{-2}$	0.99	1600	$9.96 \cdot 10^{-5}$	1.99	1602

5.4 Two-Step Runge-Kutta Methods

We now focus our attention on the family of two-step Runge-Kutta (TSRK) methods, here analyzed by using the GLMs framework described in the previous sections. TSRK methods have been introduced by Jackiewicz and Tracogna [228, 229] and rely on the following two-step formulation

$$\begin{cases} y_{n+1} = (1-\theta)y_n + \theta y_{n-1} + h\displaystyle\sum_{j=1}^{s}\left(v_j f_j^{[n]} + w_j f_j^{[n-1]}\right), \\ Y_i^{[n]} = (1-u_i)y_n + u_i y_{n-1} + h\displaystyle\sum_{j=1}^{s}\left(a_{ij} f_j^{[n]} + b_{ij} f_j^{[n-1]}\right), \quad i = 1, 2, \ldots, s, \end{cases} \tag{5.15}$$

with $f_j^{[n-1]} = f(t_{n-1}+c_j h, Y_j^{[n-1]})$ and $f_j^{[n]} = f(t_n + c_j h, Y_j^{[n]})$, $j = 1, 2, \ldots, s$. As usual, y_n is supposed to be an approximation of order p to $y(t_n)$, while the internal stage $Y_i^{[n]}$ is an approximation of stage order q to $y(t_{n-1} + c_i h)$, $i = 1, 2, \ldots, s$. TSRK methods are fully characterized by the tableau

$$\begin{array}{c|c|c} u & A & B \\ \hline \theta & v^{\intercal} & w^{\intercal} \end{array} = \begin{array}{c|ccc|ccc} u_1 & a_{11} & \cdots & a_{1s} & b_{11} & \cdots & b_{1s} \\ u_2 & a_{21} & \cdots & a_{2s} & b_{21} & \cdots & b_{2s} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ u_s & a_{s1} & \cdots & a_{ss} & b_{s1} & \cdots & b_{ss} \\ \hline \theta & v_1 & \cdots & v_s & w_1 & \cdots & w_s \end{array}$$

and admit the following compact representation

$$\begin{aligned} y_{n+1} &= (1-\theta)y_n + \theta y_{n-1} + h\left((v^\intercal \otimes I)F^{[n]} + (w^\intercal \otimes I)F^{[n-1]}\right), \\ Y^{[n]} &= ((e-u)\otimes I)y_n + (u \otimes I)y_{n-1} + h\left((A \otimes I)F^{[n]} + (B \otimes I)F^{[n-1]}\right), \end{aligned} \tag{5.16}$$

where $\otimes$ denotes the standard Kronecker tensor product, I is the identity matrix in $\mathbb{R}^{d\times d}$, $e = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}^\intercal \in \mathbb{R}^s$ and

$$Y^{[n]} = \begin{bmatrix} Y_1^{[n]} \\ Y_2^{[n]} \\ \vdots \\ Y_s^{[n]} \end{bmatrix} \in \mathbb{R}^{sd}, \quad F^{[n]} = \begin{bmatrix} f_1^{[n]} \\ f_2^{[n]} \\ \vdots \\ f_s^{[n]} \end{bmatrix} \in \mathbb{R}^{sd}, \quad F^{[n-1]} = \begin{bmatrix} f_1^{[n-1]} \\ f_2^{[n-1]} \\ \vdots \\ f_s^{[n-1]} \end{bmatrix} \in \mathbb{R}^{sd}.$$

The peculiarity of TSRK methods (5.16) lies in their dependency on the stage derivatives $F^{[n]}$ and $F^{[n-1]}$ at two consecutive subintervals: as a consequence, *"we gain extra degrees of freedom associated with a two-step scheme without the need for extra function evaluations"* [229], because the vector $F^{[n-1]}$ is completely inherited from the previous step and, therefore, the computational cost only depends on the structure of the matrix A, as for RK methods (4.8). The achieved degrees of freedom can be used, for instance, in order to improve the accuracy of existing one-step methods.

TSRK methods are multivalue methods admitting GLM representation (5.4) with $r = s + 2$ in correspondence of the vector

$$y^{[n]} = \begin{bmatrix} y_n \\ y_{n-1} \\ hF^{[n-1]} \end{bmatrix} \tag{5.17}$$

and the tableau (5.3)

$$\left[\begin{array}{c|cccc} A & e-u & u & B \\ \hline v^\intercal & 1-\theta & \theta & w^\intercal \\ 0 & 0 & 1 & 0 \\ I & 0 & 0 & 0 \end{array}\right] \in \mathbb{R}^{(2s+2)\times(2s+2)}, \tag{5.18}$$

where I is the identity matrix in $\mathbb{R}^{s\times s}$. In the remainder of this section, we assume the hypothesis of high stage order, considering TSRK methods of order p and

stage order $q = p$, i.e., we consider methods of uniform order p. Taking into account the expression of the vector $y^{[n]}$ given by (5.17), we expand $y(t_n - h)$ and $y'(t_n + (c-e)h)$ into Taylor series around t_n, obtaining

$$y^{[n]} = \begin{bmatrix} y(t_n) \\ y(t_n) - hy'(t_n) + \dfrac{h^2}{2}y''(t_n) + \cdots + (-1)^p \dfrac{h^p}{p!}y^{(p)}(t_n) \\ hy'(t_n)e + h^2(c-e)y''(t_n) + \cdots + h^p \dfrac{(c-e)^{p-1}}{(p-1)!}y^{(p)}(t_n) \end{bmatrix} + O(h^{p+1}),$$

where the power $(c-e)^\nu$, ν=0, 1, …, $p-1$, has to be intended componentwise. Then, for TSRK methods,

$$q_0 = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \end{bmatrix}^\intercal \in \mathbb{R}^{s+2}, \quad q_k = \begin{bmatrix} 0 & \dfrac{(-1)^k}{k!} & \left(\dfrac{(c-e)^{k-1}}{(k-1)!}\right)^\intercal \end{bmatrix}^\intercal,$$

$k = 1, 2, \ldots, p$. Then, the following results holds true.

Theorem 5.3 *TSRK method* (5.15) *have uniform order* p *if and only if, for any* $k = 1, 2, \ldots, p$,

$$(-1)^k u + kAc^{k-1} + kB(c-e)^{k-1} = c^k, \tag{5.19}$$

and

$$(-1)^k \theta + kv^\intercal c^{k-1} + kw^\intercal (c-e)^{k-1} = 1. \tag{5.20}$$

Proof Order conditions (5.19) and (5.20) are obtained by rewriting (5.12) and (5.13) in terms of the tableau (5.18) providing the GLM formulation of TSRK methods. In particular, (5.19) directly follows from (5.12). Equation (5.13) on the TSRK tableau (5.18) requires the computation of the vector

$$\sum_{l=0}^{k} \frac{q_{k-l}}{l!} = \begin{bmatrix} \dfrac{1}{k!} \\ \displaystyle\sum_{l=0}^{k} \dfrac{(-1)^{k-l}}{l!(k-l)!} \\ \displaystyle\sum_{l=0}^{k} \dfrac{(c-e)^{k-l-1}}{l!(k-l-1)!} \end{bmatrix}.$$

Since

$$\sum_{l=0}^{k} \frac{(-1)^{k-l}}{l!(k-l)!} = \frac{1}{k!} \sum_{l=0}^{k} \binom{k}{l} (-1)^{k-l} = 0$$

and

$$\sum_{l=0}^{k} \frac{(c-e)^{k-l-1}}{l!(k-1-l)!} = \frac{1}{(k-1)!} \sum_{l=0}^{k-1} \binom{k-1}{l} (c-e)^{k-l-1} = \frac{c^{k-1}}{(k-1)!},$$

we have

$$\sum_{l=0}^{k} \frac{q_{k-l}}{l!} = \begin{bmatrix} 1 \\ \frac{1}{k!} \\ 0 \\ \frac{c^{k-1}}{(k-1)!} \end{bmatrix}.$$

Replacing last equation and the values in (5.18) into (5.13) leads to the thesis. □

We observe that setting $k = 1$ in (5.20) leads to the consistency condition for TSRK methods

$$(v^T + w^T)e = 1 + \theta,$$

while the case $k = 1$ in (5.19) gives the condition of stage consistency

$$(A + B)e - u = c.$$

We finally analyze zero-stability, by checking the roots of the minimal polynomial of the matrix V, i.e.,

$$p(\omega) = \omega(\omega^2 - (1-\theta)\omega - \theta),$$

whose roots are $\omega = 0$, $\omega = 1$ and $\omega = -\theta$. Therefore, a TSRK is zero-stable if and only if $-1 < \theta \leq 1$.

5.5 Dense Output Multivalue Methods

According to [126], we now propose to smoothly extend a multivalue numerical method in GLM form (5.2) and depending on the Nordsieck vector (5.1) by means

of a piecewise collocation polynomial of the form

$$P_n(t_n + \vartheta h) = \sum_{i=1}^{r} \alpha_i(\vartheta) y_i^{[n]} + h \sum_{i=1}^{s} \beta_i(\vartheta) f(t_n + c_i h, P(t_n + c_i h)), \tag{5.21}$$

with $\vartheta \in [0, 1]$. This representation is provided with respect to the functional basis

$$\{\alpha_i(\vartheta),\ \beta_j(\vartheta),\ i = 1, 2, \ldots, r,\ j = 1, 2, \ldots, s\}$$

to be determined by imposing suitable conditions. In particular, since we aim to provide a collocation polynomial, we impose interpolation conditions of the type

$$P_n(t_n) = y_1^{[n]}, \quad P_n'(t_n) = y_2^{[n]}, \quad \cdots, \quad P_n^{(r-1)}(t_n) = y_r^{[n]} \tag{5.22}$$

and collocation conditions

$$P_n'(t_n + c_i h) = f(t_n + c_i h, P_n(t_n + c_i h)), \quad i = 1, 2, \ldots, s. \tag{5.23}$$

In other terms, due to the fact that derivatives up to order $r - 1$ are interpolated, the global piecewise polynomial generated by multivalue collocation is globally of class C^{r-1}. It is worth observing that most interpolants based on Runge-Kutta methods only have global C^1 continuity. The practical value of highly continuous interpolants is visible in many different situations already shown in the existing literature such as scientific visualization [253], functional differential equations with state-dependent delay [22, 202], numerical solution of differential-algebraic equations and nonlinear equations [246, 338], optimal control problems [293], discontinuous initial value problems [162, 337] or, more in general, whenever a smooth dense output is needed [207, 282].

Above interpolation conditions (5.22) on P_n are naturally reflected on the basis functions and, indeed, they are equivalent to

$$\alpha_j(0) = \delta_{j1}, \quad \alpha_j^{(\nu)}(0) = \delta_{j,\nu+1},$$

for $j = 1, 2, \ldots, r$, $\nu = 1, 2, \ldots, r - 1$ and

$$\beta_j(0) = \beta_j^{(\nu)}(0) = 0,$$

for $j = 1, 2, \ldots, s$, $\nu = 1, 2, \ldots, r-1$. Collocation conditions (5.23) are equivalent to

$$\alpha_j'(c_i) = 0, \quad i = 1, 2, \ldots, r, \quad j = 1, 2, \ldots, s,$$
$$\beta_j'(c_i) = \delta_{ij}, \quad i, j = 1, 2, \ldots, s,$$

where δ_{ij} is the Kronecker delta. Each basis function is subject to $s + r$ constraints, hence it is an algebraic polynomial of degree at most $s + r - 1$.

In summary, the collocation polynomial (5.21) is a global smooth extension of class C^{r-1} of the Nordsieck GLM (5.2) with tableau (5.3) characterized by the following matrices

$$A = \left[\beta_j(c_i)\right]_{i,j=1,\dots,s}, \quad U = \left[\alpha_j(c_i)\right]_{i=1,\dots,s,\ j=1,\dots,r},$$

$$B = \left[\beta_j^{(i-1)}(1)\right]_{i=1,\dots,r,\ j=1,\dots,s}, \quad V = \left[\alpha_j^{(i-1)}(1)\right]_{i,j=1,\dots,r}.$$

We now aim to analyze the error associated to a multivalue collocation approximant of type (5.21), relying on the assumption that it provides a uniform approximation of order p to the solution of the differential system. In other terms, a multivalue collocation polynomial (5.21) is required to satisfy

$$P_n(t_n + \vartheta h) = y(t_n + \vartheta h) + O(h^{p+1}), \qquad \vartheta \in [0, 1].$$

Then, the local discretization error associated to a single step of a multivalue collocation method can be defined as the residuum operator

$$\xi_n(t_n + \vartheta h) = y(t_n + \vartheta h) - \sum_{i=1}^{r} \alpha_i(\vartheta) h^{i-1} y^{(i-1)}(t_n) - h \sum_{i=1}^{s} \beta_i(\vartheta) y'(t_n + c_i h), \tag{5.24}$$

with $\vartheta \in [0, 1]$, and y is exact solution of the differential problem (1.1). Then, the following result holds.

Theorem 5.4 *The multivalue collocation method defined by (5.21) is an approximation of uniform order p to the solution of (1.1) if and only if*

$$\begin{aligned} &\alpha_1(\vartheta) = 1, \\ &\frac{\vartheta^\nu}{\nu!} - \alpha_{\nu+1}(\vartheta) - \sum_{i=1}^{s} \frac{c_i^{\nu-1}}{(\nu-1)!} \beta_i(\vartheta) = 0, \quad \nu = 1, 2, \dots, r-1, \\ &\frac{\vartheta^\mu}{\mu!} - \sum_{i=1}^{s} \frac{c_i^{\mu-1}}{(\mu-1)!} \beta_i(\vartheta) = 0, \quad \mu = r, \dots, p. \end{aligned} \tag{5.25}$$

Proof We expand $y(t_n + \vartheta h)$ and $y'(t_n + c_h)$ in Taylor series around t_n and replace them in (5.24), obtaining

$$\begin{aligned}\xi(t_n + \vartheta_h) = y(t_n) + \vartheta h y'(y_n) + \ldots + \frac{(\vartheta h)^p}{p!} y^{(p)}(t_n) \\ - \alpha_1(\vartheta) y(t_n) - \sum_{j=2}^{r} \alpha_i(\vartheta) h^{i-1} y^{(i-1)}(t_n) \\ - h \sum_{i=1}^{s} \beta_i(\vartheta) \left(y'(t_n) + c_i h y''(t_n) + \ldots + \frac{(c_i h)^{p-1}}{(p-1)!} y^{(p)}(t_n) \right) + O(h^{p+1}).\end{aligned}$$

Conditions (5.13) arise from annihilating all terms up to order p. □

We can then interpret conditions (5.25) as uniform order conditions for a multivalue collocation methods defined by (5.21). Moreover, from last theorem, we can also understand which is the uniform order of convergence for a multivalue collocation method.

Corollary 5.1 *The uniform order of convergence for a multivalue collocation method* (5.21) *is* $s + r - 1$.

Proof The linear system (5.25), deprived of the first identity, is a system of p linearly independent equations in $s + r - 1$ unknowns admitting a unique solution if and only if the number of equations equals that of the unknowns, i.e. when $p = s + r - 1$. □

Example 5.4 We provide an example of multivalue collocation method with $s = 1$ and $r = 2$, relying on the polynomial

$$P_n(t_n + \vartheta h) = y_1^{[n]} + \alpha_2(\vartheta) y_2^{[n]} + h\beta_1(\vartheta) f(t_n + ch, P(t_n + ch)).$$

According to Corollary 5.1, we can expect uniform order 2, which is achieved by solving conditions (5.25) for $p = 2$, i.e.

$$\begin{aligned}\vartheta - \alpha_2(\vartheta) - \beta_1(\vartheta) = 0, \\ \frac{\vartheta^2}{2} - c\beta_1(\vartheta) = 0.\end{aligned}$$

(continued)

Example 5.4 (continued)
This systems leads to

$$\alpha_2(\vartheta) = \vartheta\left(1 - \frac{\vartheta}{2c}\right), \qquad \beta_1(\vartheta) = \frac{\vartheta^2}{2c}.$$

The corresponding Butcher tableau is given by

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} \beta_1(c) & 1 & \alpha_2(c) \\ \hline \beta_1(1) & 1 & \alpha_2(1) \\ \beta_1'(1) & 0 & \alpha_2'(1) \end{array}\right] = \left[\begin{array}{c|cc} \frac{c}{2} & 1 & \frac{c}{2} \\ \hline \frac{1}{2c} & 1 & 1 - \frac{1}{2c} \\ \frac{1}{c} & 0 & 1 - \frac{1}{c} \end{array}\right].$$

For $c = \frac{1}{2}$ we obtain

$$\alpha_2(\vartheta) = \vartheta(1 - \vartheta), \qquad \beta_1(\vartheta) = \vartheta^2,$$

which is the C^1 extension of uniform order $p = 2$ of the general linear method

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} \frac{1}{4} & 1 & \frac{1}{4} \\ \hline 1 & 1 & 0 \\ 2 & 0 & -1 \end{array}\right]. \tag{5.26}$$

5.6 Exercises

1. Analyze the family of explicit GLMs (5.4) with $s = r = 2$, providing the conditions ensuring their convergence and giving examples of methods having the maximum attainable order.
2. Multistep Runge-Kutta methods

$$y_{n+1} = \sum_{i=1}^{k} v_j y_{n+1-i} + h \sum_{i=1}^{s} b_i f(t_n + c_i h, Y_i),$$

$$Y_i = \sum_{j=1}^{k} u_{ij} y_{n+1-j} + h \sum_{j=1}^{s} a_{ij} f(t_n + c_j h, Y_j)$$

have been introduced by Burrage in [43, 44] as multistep extension of RK methods (4.8). Write them as GLMs (5.4), give the expression of the vector $y^{[n]}$ and analyze their convergence by the results on GLMs provided in this chapter.

3. Compute the vectors q_k in (5.12) and (5.13) for GLMs (5.4) where the vector $y^{[n]}$ is given by the Nordsieck vector (5.1).
4. Provide the convergence analysis of the following TSRK method (5.15) depending on the Butcher tableau

$$\begin{array}{c|c|c} u & A & B \\ \hline \theta & v^{\intercal} & w^{\intercal} \end{array} = \begin{array}{c|cc} \frac{-3+\sqrt{6}}{6} & \frac{1}{2} & \frac{\sqrt{6}}{6} \\ \hline 0 & \frac{3-\sqrt{6}}{6} & \frac{3+\sqrt{6}}{6} \end{array} \tag{5.27}$$

in two ways: by regarding the method as GLM and applying the convergence result of GLMs; by direct using the results for the convergence of TSRK methods.

5. Write a code in your favorite programming language implementing the TSRK method (5.27), for the numerical solution of d-dimensional systems (1.1). You need to recover the missing starting value y_1 by a chosen starting method. Provide an experimental evidence of the convergence of the method and of its order. How does the choice of the starting method affect the accuracy of the TSRK method?
6. Analyze the order and the stage-order of the GLM (5.26). Do they confirm the second order of convergence of the corresponding collocation based method, derived in Example 5.4?
7. Which is the relationship between the set of uniform order conditions (5.25) of the collocation based method depending on the polynomial (5.21) and the set of order conditions (5.13) for GLMs (5.4) where $y^{[n]}$ is the Nordsieck vector (5.1)?
8. Recast the arguments in Sect. 5.5, supposing that the functional basis is not constituted by algebraic polynomials, but by trigonometric functions. The development of dense output GLMs relying on mixed basis functions is object of [96].
9. Rewrite Program (5.1), supposing that the Jacobian of the vector field is not exactly computed, but approximated by finite differences.
10. Recast the following family of numerical methods for the solution of (1.1)

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

as GLMs (5.2), analyzing their convergence. The matrices characterizing this family of methods are $A, B, R \in \mathbb{R}^{s\times s}$, $\otimes$ denotes the usual Kronecker tensor product and $I \in \mathbb{R}^{d\times d}$ is the identity matrix. The general i-th component of the vector $Y^{[n]}$ approximates the solution of (1.1) in the internal point $t_n + c_i h$. The vector evaluations of the vector field in the entries of $Y^{[n]}$ are collected in the vector $F^{[n]} = \left[f(t_n + c_i h, Y_i^{[n]})\right]_{i=1}^{s}$. These methods are known in the literature as *peer methods* [311] and they only share the step-by-step approximations related to the internal stages.

Chapter 6
Linear Stability

A method which cannot handle satisfactorily the linear test system is not a suitable candidate for incorporation into an automatic code. More precisely, linear stability theory provides a useful yardstick (if one can have a yardstick in the complex plane!) by which different linear multistep methods (or classes of such methods) can be compared as candidates for inclusion in an automatic code.

(John D. Lambert [242])

So far, we have studied properties of numerical methods for (1.1) mostly occurring when the stepsize tends to 0: for instance, the local accuracy property of consistency, providing the coherence between the solution of difference equation and that of the corresponding differential problem under the localizing assumption; or the global accuracy property of zero-stability, ensuring that the difference between exact and numerical solutions does not blow-up as the stepsize goes to 0; and, finally, the request for convergence, guaranteeing that the global error goes to 0, when the stepsize tends to 0. Clearly, characteristic properties occurring when the stepsize goes to 0 do not reveal much of what happens for fixed values of the stepsize. In this section we handle this issue, presenting the so-called *linear stability* theory.

6.1 Dahlquist Test Equation

The theory of linear stability is the analysis of the behavior of a given class of methods applied to the so-called *Dahlquist test equation*, given by the linear scalar problem

$$y'(t) = \lambda y(t), \quad \lambda \in \mathbb{C}, \quad \mathrm{Re}(\lambda) < 0. \tag{6.1}$$

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_6

According to Theorem 1.8, the problem described by (6.1) is asymptotically stable, but we can have a direct confirmation of this issue by checking its analytical solution

$$y(t) = ce^{\lambda t}, \quad c \in \mathbb{R}, \tag{6.2}$$

that exponentially decays as t tends to infinity. Although very simple, the test problem (6.1) is able to reveal relevant properties of a numerical method, as clearly highlighted by Germund Dahlquist in his foundational paper [108].

Once a method for (1.1) is applied to the linear test equation (6.1), a natural question is the following: which values of the stepsize ensure that the asymptotic stability of the solution to (6.1) is also inherited by the numerical solution? In this sense, linear stability analysis is the study of a conservation property, i.e., the preservation of the monotonicity of (6.2) along the discretized dynamics associated to the numerical solution of (6.1) computed by a given numerical method.

Let us now focus on the connection between the general ODE problem (1.1) and the Dahlquist test problem (6.1). To do this, let $\varphi(t)$ be a smooth solution of (1.1) corresponding to a given initial value; correspondingly, we compute the linearization of the vector field in (1.1) around $\varphi(t)$, given by

$$y'(t) = f(t, \varphi(t)) + J(t, \varphi(t))(y(t) - \varphi(t)) + \text{higher order terms},$$

where J is the Jacobian of the vector field f in (1.1). By denoting $\overline{y}(t) = y(t) - \varphi(t)$, we obtain

$$\overline{y}'(t) = J(t, \varphi(t))\overline{y}(t) + \text{higher order terms}.$$

Denoting by $A \in \mathbb{R}^{d \times d}$ the frozen Jacobian at time t, the linearized version of (1.1) then assumes the form

$$z'(t) = Az(t). \tag{6.3}$$

If the matrix A has d distinct eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_d$, there exists an invertible matrix Q such that $Q^{-1}AQ = \Lambda$, where Λ is the diagonal matrix

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_d \end{bmatrix}.$$

Then, system (6.3) assumes the form

$$\overline{z}'(t) = \Lambda \overline{z}(t),$$

with $\overline{z}(t) = Q^{-1}z(t)$. Last equation defines a system of d linear scalar uncoupled equations of the type

$$\overline{z}_i{}'(t) = \lambda_i \overline{z}_i(t), \quad i = 1, 2, \ldots, d,$$

whose right-hand side is the same as that in Dahlquist test equation (6.1). In summary, Dahlquist test equation is recovered by linearization of the vector field of (1.1). However, even if (6.1) provides a remarkable simplification of (1.1), it is a useful tool to highlight meaningful properties which are relevant, for instance, in the numerical solution of stiff problems, described in Chap. 7.

6.2 Absolute Stability of Linear Multistep Methods

Let us apply a LMM (3.1) to the Dahlquist test problem (6.1), obtaining

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\lambda \sum_{j=0}^{k} \beta_j y_{n+j}.$$

By denoting $\widehat{h} = h\lambda$, we have

$$\sum_{j=0}^{k} (\alpha_j - \widehat{h}\beta_j) y_{n+j} = 0, \tag{6.4}$$

which is a scalar homogeneous linear difference equation. As observed in Sect. 2.2, solving homogeneous linear difference equations (2.8) requires the computation of the zeros of the corresponding characteristic polynomial (2.13) that, for (6.4), assumes the form

$$\sum_{j=0}^{k} (\alpha_j - \widehat{h}\beta_j) z^j = 0.$$

In terms of first and second characteristic polynomials (3.14) and (3.15), last equation is equivalent to

$$\rho(z) - \widehat{h}\sigma(z) = 0.$$

Definition 6.1 For a given LMM (3.1), the *stability polynomial* is given by

$$\pi(z, \widehat{h}) = \rho(z) - \widehat{h}\sigma(z), \tag{6.5}$$

where $\rho(z)$ and $\sigma(z)$ are the first and second characteristic polynomials (3.14) and (3.15) of (3.1).

In order to reproduce the same behavior of the exact solution of (6.1) along the solutions computed by (6.4), we need to require that the stability polynomial (6.5) satisfies the root condition introduced in Definition 3.8, for given values of $\widehat{h} \in \mathbb{C}$. This fact motivates the following definition.

Definition 6.2 A linear multistep method (3.1) is *absolutely stable* for a given $\widehat{h} \in \mathbb{C}$, if the stability polynomial (6.5) satisfies the root condition introduced in Definition 3.8.

Definition 6.3 For a linear multistep method (3.1), the *region of absolute stability* is the set

$$\mathcal{R} = \{\widehat{h} \in \mathbb{C} : \pi(z, \widehat{h}) \text{ satisfies the root condition}\}. \tag{6.6}$$

Definition 6.4 For a linear multistep method (3.1), the *stability interval* is the intersection of the stability region (6.6) with the real axis.

In summary, the root condition applied to the stability polynomial $\pi(z, \widehat{h})$ guarantees that the numerical solution computed by the corresponding LMM (3.1) applied to (6.1) reproduces the same behavior of its exact solution, for any $\widehat{h}$ belonging to the stability region (6.6). Hence, $\widehat{h} \in \mathcal{R}$ is the discrete counterpart of $\mathrm{Re}(\lambda) < 0$ for linear multistep methods (3.1).

Example 6.1 Let us compute the stability region of Euler method (2.19). Regarded as LMM, its first characteristic polynomial (3.14) is

$$\rho(z) = z - 1,$$

while its second characteristic polynomial (3.15) is

$$\sigma(z) = 1.$$

Then its stability polynomial (6.5) is given by

$$\pi(z, \widehat{h}) = z - 1 - \widehat{h}$$

and its root $z = 1 + \widehat{h}$ has to satisfy the condition

$$|1 + \widehat{h}| \leq 1,$$

describing the circle of unitary radius and centered in $(-1, 0)$, which is the stability region of the explicit Euler method. Its stability interval is then given by $[-2, 0]$.

Example 6.2 In the previous example, we have computed the stability interval of Euler method (2.19), equal to $[-2, 0]$. We now aim to understand the meaning of this issue. Euler method is absolutely stable, according to Definition 6.2, for any $\widehat{h} \in \mathbb{C}$ whose real part belongs to the interval $[-2, 0]$. Equivalently, $-2 \leq h\text{Re}(\lambda) \leq 0$, i.e.,

$$0 \leq h \leq -\frac{2}{\text{Re}(\lambda)}. \tag{6.7}$$

In other terms, a bounded stability interval imposes a stepsize restriction. Such a restriction is only due to stability purposes and it is not related to the accuracy we aim to achieve. In other terms, a value of h outside the interval $[0, -\frac{2}{\text{Re}(\lambda)}]$ provides unstable numerical solutions by using Euler method (2.19); this is not the case if the constraint (6.7) is satisfied. A numerical evidence is given by applying Euler method (2.19) to the test problem

$$\begin{aligned} y'(t) &= -4y(t), \quad t \in [0, 10], \\ y(0) &= 1, \end{aligned}$$

(continued)

Example 6.2 (continued)
whose exact solution is $y(t) = \mathrm{e}^{-4t}$. In this case, the restriction (6.7) to the stepsize becomes

$$0 \leq h \leq \frac{1}{2}.$$

Applying the method with stepsize $\frac{1}{10}$ leads to an error in the final point equal to 4.2483e−18, confirming that the numerical solution has inherited the stable behavior of the exact solution. This is not the case when the chosen stepsize is $\frac{2}{3}$: in this case the error in the final point is 2.1268e+03 and the numerical solution is not stable. It is interesting to analyze what happens when the stepsize is $\frac{1}{2}$. In this case the error in the last point is big, since it is equal to 1, but the solution is not unstable: this confirms that the stepsize restriction (6.7) has to be respected only for stability purposes, but it is not responsible of the accuracy of the method.

Example 6.3 Let us study the stability properties of Milne-Simpson method (3.2), applied to Dahlquist test problem (6.1). We obtain

$$\left(1 - \frac{\widehat{h}}{3}\right) y_{n+2} - \frac{4}{3}\widehat{h} y_{n+1} - \left(1 + \frac{\widehat{h}}{3}\right) y_n = 0, \tag{6.8}$$

i.e., a second-order homogeneous linear difference equation. Let us solve it, using the arguments treated in Chap. 2, by first computing the roots of its characteristic polynomial as solutions of the algebraic equation

$$\left(3 - \widehat{h}\right) x^2 - 4\widehat{h}x - \left(3 + \widehat{h}\right) = 0,$$

given by

$$x_1 = \frac{2\widehat{h} + \sqrt{9 - 3\widehat{h}^2}}{3 - \widehat{h}}, \quad x_2 = \frac{2\widehat{h} - \sqrt{9 - 3\widehat{h}^2}}{3 - \widehat{h}}.$$

Then, the solution of Eq. (6.8) is given by

$$y_n = \sigma_1 x_1^n + \sigma_2 x_2^n, \quad \sigma_1, \sigma_2 \in \mathbb{R}.$$

(continued)

Example 6.3 (continued)
The reader can check that

$$x_1 = 1 + \widehat{h} + O(\widehat{h}^2) = \exp(\widehat{h}) + O(\widehat{h}^2),$$

$$x_2 = -1 + \frac{\widehat{h}}{3} + O(\widehat{h}^2) = \exp\left(-\frac{\widehat{h}}{3}\right) + O(\widehat{h}^2).$$

As a consequence,

$$y_n \approx \sigma_1 \exp\left(\lambda(t_n - t_0)\right) + \sigma_2 \exp\left(-\frac{\lambda(t_n - t_0)}{3}\right)$$

and, assuming $\mathrm{Re}(\lambda) < 0$, its first summand tends to 0 when t_n grows, while the second one exhibits an exponential growth in t_n. The term associated to the root x_2 is denoted in the literature as *parasitic component*. These components destroy the overall accuracy (and, especially, the long-term behavior) of the underlying numerical method, so they deserve a special attention. We dedicate our efforts in understanding the role of parasitic components in Chap. 8, in also in order to understand when their exponential blow-up becomes visible in the numerical dynamics.

6.3 Absolute Stability of Runge-Kutta Methods

Let us now move to the linear stability analysis of Runge-Kutta methods (4.11). Such methods, applied to the Dahlquist test equation (6.1), assume the form

$$y_{n+1} = y_n + \widehat{h} b^\intercal Y,$$

$$Y = e y_n + \widehat{h} A Y.$$

The second equation is equivalent to

$$Y = (I - \widehat{h}A)^{-1} e y_n,$$

where $I \in \mathbb{R}^{s \times s}$ is the identity matrix. As a consequence,

$$y_{n+1} = \left(1 + \widehat{h} b^\intercal (I - \widehat{h}A)^{-1} e\right) y_n.$$

Then, we give the following definition.

Definition 6.5 For a given RK method (4.11), its *stability function* is defined as follows:

$$R(\widehat{h}) = 1 + \widehat{h}b^{\intercal}(I - \widehat{h}A)^{-1}e. \tag{6.9}$$

In other terms, a RK method (4.11) applied to (6.1) assumes the form

$$y_{n+1} = R(\widehat{h})y_n \tag{6.10}$$

and, as a consequence, the monotonicity of the solution to (6.1) is inherited by its numerical approximation computed by a RK method if $|R(\widehat{h})| < 1$, for given values of $\widehat{h} \in \mathbb{C}$. This motivates the following definition.

Definition 6.6 A Runge-Kutta method (4.11) is *absolutely stable* for a given $\widehat{h} \in \mathbb{C}$, if the stability function (6.9) satisfies the condition

$$|R(\widehat{h})| < 1. \tag{6.11}$$

Definition 6.7 For a Runge-Kutta method (4.11), the *region of absolute stability* is the set

$$\mathcal{R} = \{\widehat{h} \in \mathbb{C} : |R(\widehat{h})| < 1\}. \tag{6.12}$$

Definition 6.8 For a Runge-Kutta method (4.11), the *stability interval* is the intersection of the stability region (6.12) with the real axis.

In summary, condition (6.11) on the stability function $R(\widehat{h})$ guarantees that the numerical solution computed by the corresponding RK method (4.11) applied to (6.1) reproduces the same behavior of its exact solution, for any $\widehat{h}$ belonging to the stability region (6.12). Hence, in terms of stability of exact and approximate solutions, $|R(\widehat{h})| < 1$ is the discrete counterpart of $\mathrm{Re}(\lambda) < 0$ for RK methods.

Example 6.4 Let us compute the stability region of the Gaussian RK method (4.24). The corresponding stability function (6.9) is given by

$$R(\widehat{h}) = \frac{2+\widehat{h}}{2-\widehat{h}}.$$

In this case, the stability condition (6.11) holds true for any $\widehat{h} \in \mathbb{C}$ such that $\mathrm{Re}(\widehat{h}) < 0$. Then, the stability region (6.12) is the whole negative half plane and the stability interval is given by $(-\infty, 0]$.

An equivalent expression for the stability function (6.9) has been provided by Dekker and Verwer (Heerhugowaard, 1946-Heiloo, 2011) in [141], according to the following result (also see [67]).

Theorem 6.1 *The stability function (6.9) of a Runge-Kutta method admits the form*

$$R(\widehat{h}) = \frac{\det(I + \widehat{h}(eb^{\intercal} - A))}{\det(I - \widehat{h}A)}. \tag{6.13}$$

Proof For a given couple of vectors $u, v \in \mathbb{R}^s$, the determinant of the matrix $I + uv^{\intercal}$ is given by $1 + v^{\intercal}u$ (the proof is left to the reader). As a consequence

$$\det(I + \widehat{h}eb^{\intercal}(I - \widehat{h}A)^{-1}) = 1 + \widehat{h}b^{\intercal}(I - \widehat{h}A)^{-1}e = R(\widehat{h}).$$

Since

$$I + \widehat{h}(eb^{\intercal} - A) = (I + \widehat{h}eb^{\intercal}(I - \widehat{h}A)^{-1})(I - \widehat{h}A),$$

we have

$$\det(I + \widehat{h}(eb^{\intercal} - A)) = R(\widehat{h}) \det(I - \widehat{h}A)$$

and the thesis holds true. □

Equation (6.13) is useful to realize that the stability function of an explicit RK method is an algebraic polynomial in $\widehat{h}$, since $\det(I - \widehat{h}A) = 1$. As a consequence, the corresponding stability region (6.12) is necessarily bounded. For implicit RK methods, the stability function (6.13) is a rational function and, in this case, unbounded stability regions are allowed.

We finally observe that there is a connection between the stability function (6.9) and the rational approximation of the exponential function. Indeed, replacing the exact solution (6.2) with $c = 1$ in the recurrence (6.10) and denoting by p the order of the corresponding RK method, we obtain

$$\mathrm{e}^{\lambda(t_n+h)} = R(\widehat{h})\mathrm{e}^{\lambda t_n} + O(h^{p+1}),$$

or, equivalently,

$$\mathrm{e}^{\lambda t_n}(\mathrm{e}^{\widehat{h}} - R(\widehat{h})) = O(h^{p+1}).$$

Last equation holds true for any t_n if

$$R(\widehat{h}) = \mathrm{e}^{\widehat{h}} + O(h^{p+1}).$$

In other terms, $R(\widehat{h})$ is an approximation of order p to the exponential $\mathrm{e}^{\widehat{h}}$.

6.4 Absolute Stability of Multivalue Methods

Let us now provide the linear stability analysis of multivalue methods in GLMs form (5.4) (also refer to [63, 67, 228]). Such methods, applied to the Dahlquist test equation (6.1), read

$$Y = \widehat{h}AY + Uy^{[n-1]},$$

$$y^{[n]} = \widehat{h}BY + Vy^{[n-1]}$$

and the first equation is equivalent to

$$Y = (I - \widehat{h}A)^{-1}Uy^{[n-1]},$$

where $I \in \mathbb{R}^{s\times s}$ is the identity matrix. As a consequence,

$$y^{[n]} = \left(V + \widehat{h}B(I - \widehat{h}A)^{-1}U\right) y^{[n-1]}.$$

Definition 6.9 For a given multivalue method (5.4), its *stability matrix* is defined by

$$S(\widehat{h}) = V + \widehat{h}B(I - \widehat{h}A)^{-1}U. \tag{6.14}$$

In other terms, a multivalue method (5.4) applied to (6.1) assumes the form

$$y^{[n]} = S(\widehat{h})y^{[n-1]}.$$

As a consequence, the monotonicity of the solution to (6.1) is inherited by its numerical approximation computed by a multivalue method if the spectral radius of the stability matrix $\rho(S(\widehat{h})) < 1$, for given values of $\widehat{h} \in \mathbb{C}$. This motivates the following definition.

Definition 6.10 A multivalue method (5.4) is *absolutely stable* for a given $\widehat{h} \in \mathbb{C}$, if the stability matrix (6.14) satisfies the condition

$$\rho(S(\widehat{h})) < 1, \tag{6.15}$$

where $\rho(S(\widehat{h}))$ is the spectral radius of the stability matrix (6.14).

Definition 6.11 For a multivalue method (5.4), the *region of absolute stability* is the set

$$\mathcal{R} = \{\widehat{h} \in \mathbb{C} : \rho(S(\widehat{h})) < 1\}. \tag{6.16}$$

Definition 6.12 For a multivalue method (5.4), the *stability interval* is the intersection of the stability region (6.16) with the real axis.

In summary, condition (6.15) guarantees that the numerical solution computed by the corresponding multivalue method (4.11) applied to (6.1) reproduces the same behavior of its exact solution, for any $\widehat{h}$ belonging to the stability region (6.16). Hence, in terms of stability of exact and approximate solutions, $\rho(S(\widehat{h})) < 1$ is the discrete counterpart of $\mathrm{Re}(\lambda) < 0$ for multivalue methods.

Example 6.5 Let us compute the stability region of the Gaussian RK method (4.24) using its multivalue representation (5.4), depending on the Butcher tableau

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|c} \frac{1}{2} & 1 \\ \hline 1 & 1 \end{array}\right].$$

The corresponding stability matrix (6.14) is then given by the scalar

$$S(\widehat{h}) = \frac{2+\widehat{h}}{2-\widehat{h}},$$

confirming the analogous analysis given in Example 6.4. Then, the stability region (6.16) is the whole negative half plane and the stability interval is given by $(-\infty, 0]$.

We finally observe that the analysis of the stability matrix (6.14) and its characteristic polynomial may be a non-trivial problem, especially when a large number of internal stages is involved. For this reason, techniques in the direction of remarkably reducing the complexity of this issue have been introduced in the literature. In particular, for a multivalue method (5.4), a desired property is the so-called *inherent Runge-Kutta stability*, i.e., the characteristic polynomial $p(\omega, \widehat{h})$ of the stability matrix (6.14) can be factor out as

$$p(\omega, \widehat{h}) = \omega^{r-1}(\omega - R(\widehat{h})),$$

being $R(\widehat{h})$ the stability function (6.9) of a Runge-Kutta method. In this way, linear stability analysis of multivalue methods with inherent Runge-Kutta stability can be remarkably simplified, since it only relies on the analysis of the properties of $R(\widehat{h})$. The construction of multivalue methods in the form of GLMs (5.4) with inherent Runge-Kutta stability has been addressed, for instance, in [67, 228, 348] and references therein.

6.5 Boundary Locus

We now aim to discuss a technique useful to draw the stability region of a numerical method, based on the plot of its boundary $\partial\mathcal{R}$. For this reason, the procedure we are going to present is known in the literature as *boundary locus* technique [242].

Let us first analyze the boundary locus of a linear multistep method (3.1). According to Definition 6.3, the boundary of the stability region of a LMM is given by the values $\widehat{h} \in \mathbb{C}$ such that there exists at least a root of the corresponding stability polynomial (6.5) having unitary modulus. Such a root is then of the form $e^{i\theta}$, with $\theta \in [0, 2\pi]$ and, since it is solution of (6.5), we have

$$\pi(e^{i\theta}, \widehat{h}) = 0.$$

Therefore,

$$\widehat{h}(\theta) = \frac{\rho(e^{i\theta})}{\sigma(e^{i\theta})}, \quad \theta \in [0, 2\pi], \tag{6.17}$$

that is the analytic expression of a parametric curve describing the boundary of the stability region of the corresponding LMM.

We now provide a Matlab implementation of the boundary locus technique for LMMs (3.1).

Program 6.1 (Boundary Locus of a Linear Multistep Method)

```
% Function drawing the boundary locus of a given LMM

% Inputs:
% - alf, vector of the coefficients α0, α1, ..., αk;
% - bet, vector of the coefficients β0, β1, ..., βk;

% Output:
% plot of the boundary locus of the corresponding LMM

function boundaryLocusLMM(alf,bet)
theta=linspace(0,2*pi);
r=exp(1i*theta);
h=polyval(alf,r)./polyval(bet,r);
plot(real(h),imag(h))
```

Example 6.6 Let us draw the stability regions of the following linear multistep methods:

- the explicit Euler method (2.19), requiring as inputs

$$\texttt{alf} = [-1 \;\; 1], \quad \texttt{bet} = [1 \;\; 0];$$

- the implicit Euler method (2.32), with inputs

$$\texttt{alf} = [-1 \;\; 1], \quad \texttt{bet} = [0 \;\; 1];$$

(continued)

Example 6.6 (continued)
- the trapezoidal method (2.33), with inputs

$$\texttt{alf} = [-1 \;\; 1], \quad \texttt{bet} = [1/2 \;\; 1/2];$$

- the two-step Adams-Bashforth method (3.3), with inputs

$$\texttt{alf} = [0 \;\; -1 \;\; 1], \quad \texttt{bet} = [-1/2 \;\; 3/2 \;\; 0].$$

Figures 6.1, 6.2, 6.3, and 6.4 show the stability regions of above methods, shaded in grey. We can observe that explicit Euler and the two-step Adams-Bashforth methods have bounded stability regions; implicit Euler and trapezoidal methods have unbounded stability regions.

Let us now provide the boundary locus of selected RK methods (4.8). According to Definition 6.7, the boundary of the stability region of RK methods is given by the values $\widehat{h} \in \mathbb{C}$ such that $R(\widehat{h}) = \mathrm{e}^{i\theta}$, with $\theta \in [0, 2\pi]$, i.e., the modulus of the stability function is equal to 1. For a possible implementation of the boundary locus technique for RK methods, see Exercise 2.

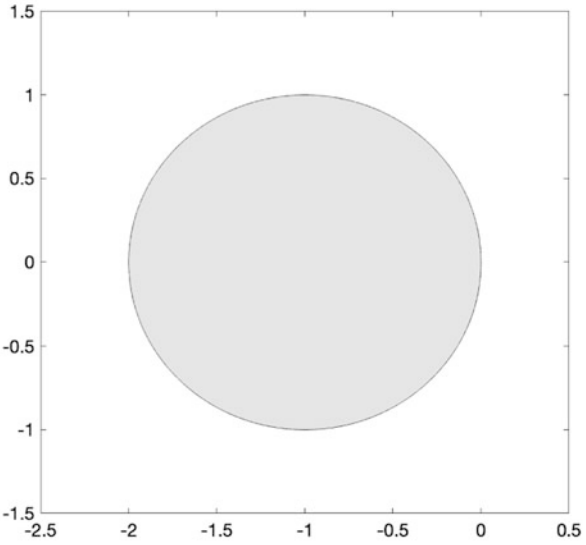


Fig. 6.1 Stability region of the explicit Euler method (2.19)

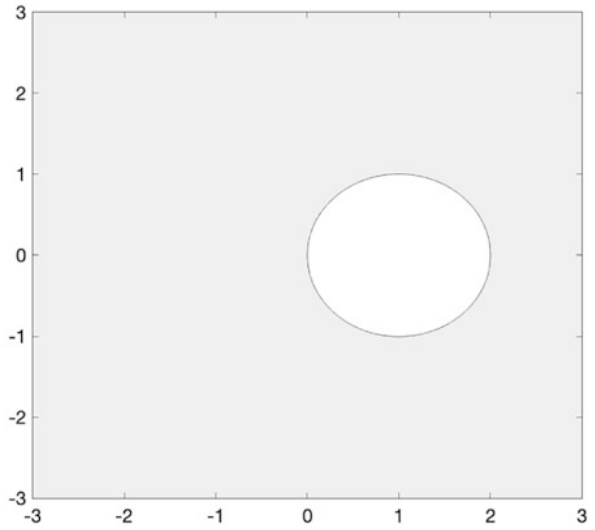


Fig. 6.2 Stability region of the implicit Euler method (2.32)

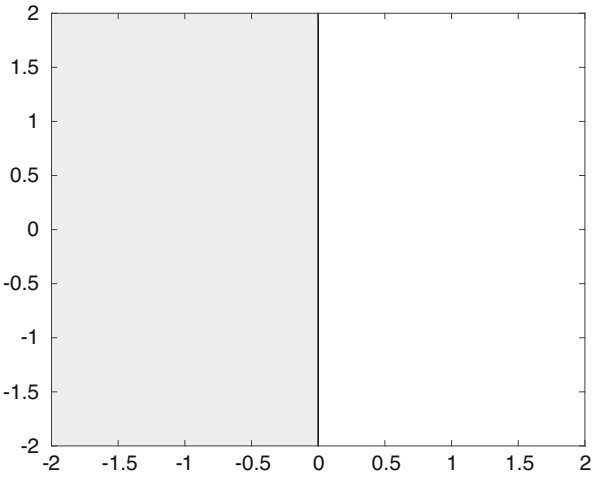


Fig. 6.3 Stability region of the trapezoidal method (2.33)

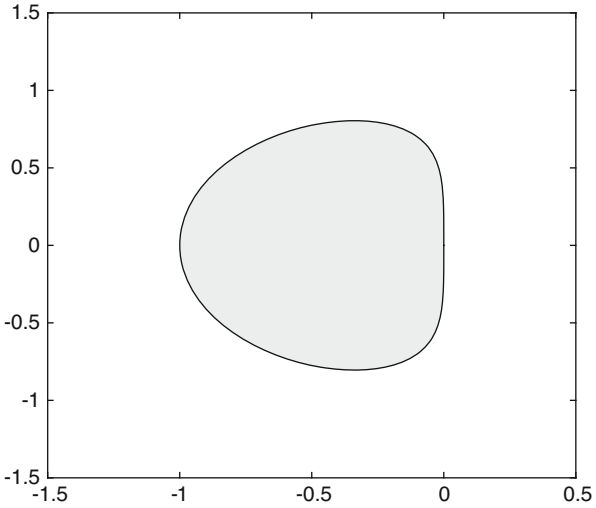


Fig. 6.4 Stability region of two-step Adams-Bashforth method (3.3)

Example 6.7 Let us draw the stability region of the 3/8-method (4.21), whose stability function is given by

$$R(\widehat{h}) = 1 + \widehat{h} + \frac{1}{2}\widehat{h}^2 + \frac{1}{6}\widehat{h}^3 + \frac{1}{24}\widehat{h}^4.$$

Figure 6.5 shows the corresponding stability region, shaded in grey. As expectable, such a region is bounded, since the 3/8-method is explicit and its stability function is an algebraic polynomial.

Example 6.8 Let us now display the stability region of the two-stage Radau IA method (4.26), whose stability function is given by

$$R(\widehat{h}) = \frac{2(\widehat{h}+3)}{\widehat{h}^2 - 4\widehat{h} + 6}.$$

Figure 6.6 shows the corresponding stability region, shaded in grey. The region is unbounded: as discussed in Sect. 6.3, since the method is implicit, $R(\widehat{h})$ is a rational function and, correspondingly, unbounded stability regions are admitted.

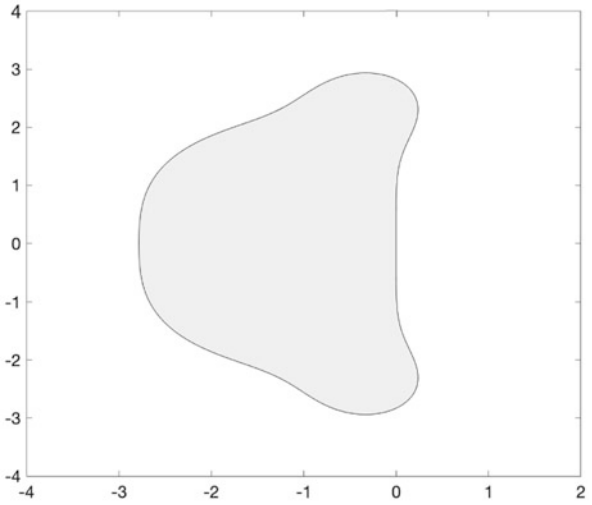


Fig. 6.5 Stability region of the 3/8-method (4.21)

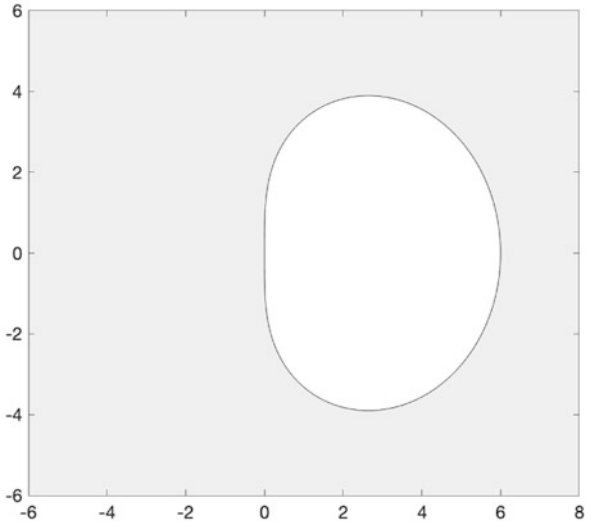


Fig. 6.6 Stability region of the two-stage Radau IA method (4.26)

We finally analyze the stability regions of multivalue methods (5.4). According to Definition 6.10, the boundary of the stability region of multivalue methods is given by the values $\widehat{h} \in \mathbb{C}$ such that the spectral radius of the stability matrix (6.14) is equal to 1. For a possible implementation of the boundary locus technique for multivalue methods, see Exercise 3.

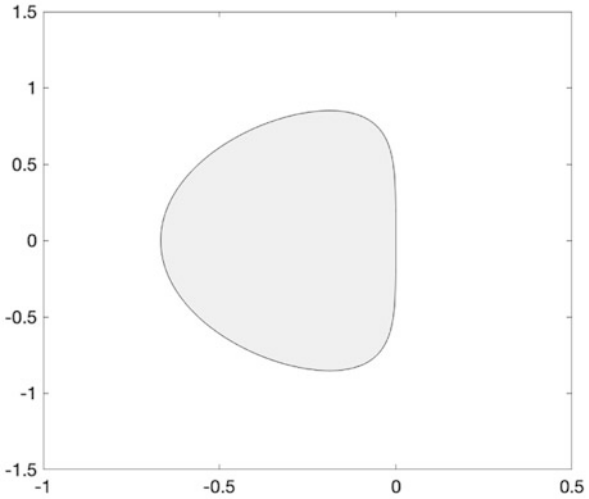


Fig. 6.7 Stability region of the multivalue method GLM2 (5.14)

Example 6.9 We draw the stability region of the multivalue method GLM2 (5.14), whose stability matrix is given by

$$S(\widehat{h}) = \begin{bmatrix} \frac{\widehat{h}}{6} - \frac{1}{3} & \frac{4}{3} - \frac{\widehat{h}}{2} \\ -\frac{\widehat{h}}{2} & \frac{3}{2}\widehat{h} + 1 \end{bmatrix}.$$

Figure 6.7 shows the corresponding stability region, shaded in grey. The region is bounded, as expectable since the method is explicit.

6.6 Unbounded Stability Regions

In the previous section we have seen that stability regions may be bounded (as it is always the case for explicit methods) or unbounded (as it may happen for implicit methods). We now aim to focus our attention on unbounded stability regions, giving some relevant definitions.

6.6.1 A-Stability

The first stability notion we introduce brings to the inclusion of the stability domain of the test problem (6.1), i.e., the set of points in the complex plane with negative real part, in the stability region of a numerical method applied to (6.1). Such an issue has been introduced by Dahlquist in his famous paper [108] (also see [66]).

Definition 6.13 A numerical method for (1.1) is *A-stable* if its stability region contains the stability domain of (6.1), i.e., the set of points in the complex plane with negative real part.

Hence, the stability region of an A-stable method certainly contains the left half plane displayed in Fig. 6.8. As a consequence, according to the examples provided in the previous section, the implicit Euler method (2.32), the trapezoidal method (2.33) and the Radau IA method are certainly A-stable.

A curiosity about the choice of this denomination (i.e., A-stability) comes from Dahlquist himself, through the following quotation reported in [195]: *"I didn't like all these "strong", "perfect", "absolute", "generalized", "super", "hyper", "complete" and so on in mathematical definitions, I wanted something neutral; and having been impressed by David Young's "property A", I choose the term A-stable"*.

Alternative stability definitions, though weaker than A-stability, are based on providing unbounded stability intervals, according to the following definitions.

Fig. 6.8 (Shaded) region of the complex plane contained in the stability region of an A-stable method

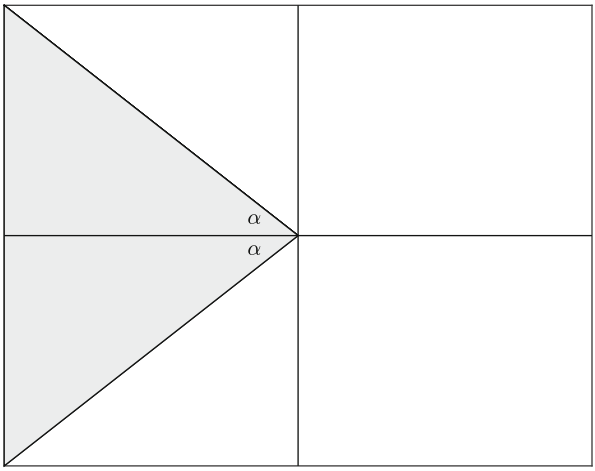


Fig. 6.9 (Shaded) region of the complex plane contained in the stability region of an A(α)-stable method

Definition 6.14 A numerical method for (1.1) is *A(α)-stable*, for $\alpha \in (0, \pi/2)$, if its stability region contains the set $\{\widehat{h} \in \mathbb{C} : -\alpha < \pi - \arg \widehat{h} < \alpha\}$.

According to this definition, introduced by Widlund in [344], the stability region of an A(α)-stable numerical method contains the sector shaded in Fig. 6.9. The stability interval of an A(α)-stable method is the whole negative real axis, as it is for A-stable methods. We also observe that A-stability can also be defined as A($\pi/2$)-stability.

We conclude with the following definition, provided by Cryer in [105].

Definition 6.15 A numerical method for (1.1) is *A_0-stable* if its stability region contains the negative real axis.

According to above definitions, all A-stable methods are also A(α)-stable and A_0-stable; A(α)-stable methods are also A_0-stable.

6.6.2 Padé Approximations

As seen in Sect. 6.3, the stability function of a Runge-Kutta method is a rational approximation of order p to the exponential. In some sense, we can argue that there

is a connection between the stability properties of Runge-Kutta methods and rational approximations to the exponential. Hence, we are interested in analyzing rational functions of the type

$$R_m^n(x) = \frac{\sum_{i=0}^{n} a_i x^i}{\sum_{j=0}^{m} b_j x^j}, \quad x \in \mathbb{C}, \tag{6.18}$$

where n and m are given non-negative integer numbers. We always suppose that $a_0 = b_0 = 1$, $a_n \neq 0$ and $b_m \neq 0$. The function $R_m^n(x)$ is a *rational approximation of order p to the exponential* if $R_m^n(x) = \mathrm{e}^x + O(x^{p+1})$ or, equivalently, if

$$\sum_{i=0}^{n} a_i x^i - \left(\sum_{j=0}^{m} b_j x^j\right)\left(\sum_{k=0}^{\infty} \frac{x^k}{k!}\right) = O(x^{p+1}). \tag{6.19}$$

Example 6.10 Let us construct the approximant $R_1^1(x)$ to the exponential, of the type

$$R_1^1(x) = \frac{1 + a_1 x}{1 + b_1 x}.$$

According to Eq. (6.19), we have

$$1 + a_1 x - (1 + b_1 x)\left(1 + x + \frac{x^2}{2} + \ldots\right) = O(x^{p+1}).$$

Collecting the powers of x leads to

$$(a_1 - b_1 - 1)x - \left(b_1 + \frac{1}{2}\right)x^2 = O(x^3),$$

which gives

$$a_1 = \frac{1}{2}, \quad b_1 = -\frac{1}{2}.$$

Hence,

$$R_1^1(x) = \frac{1 + \frac{1}{2}x}{1 - \frac{1}{2}x}.$$

The rational function (6.18) depends on $n + m$ unknown coefficients that solve $n + m$ algebraic conditions obtained by annihilating the first $n + m$ terms in the left-hand side of (6.19). Such terms are the coefficients of $x, x^2, \ldots, x^{n+m}$ and, as a consequence, the right-hand side of (6.19) is $O(x^{n+m+1})$. Hence, the maximal attainable order is $p = n + m$. Maximal order rational approximations to the exponential are called *Padé approximations*.

The following theorem, proved by Butcher, gives the coefficients a_i and b_j of the Padé approximation (6.18) in closed form. The interested reader can find the proof in [67].

Theorem 6.2 *The coefficients of the Padé approximation* R_m^n (6.18) *are given by*

$$a_i = \frac{n!}{(n+m)!}\frac{(n+m-i)!}{i!(n-i)!}, \quad i = 1, 2, \ldots, n,$$

$$b_j = (-1)^j \frac{m!}{(n+m)!}\frac{(n+m-j)!}{i!(m-j)!}, \quad j = 1, 2, \ldots, m.$$

The following definition, provided by Ehle [156, 157], is relevant in creating a bridge among rational approximations to the exponential and stability.

Definition 6.16 A rational approximation to the exponential $R_m^n(x)$ is said *A-acceptable* if $|R_m^n(x)| < 1$, for any $x \in \mathbb{C}$ such that $\mathrm{Re}(x) < 0$.

As a consequence, a Runge-Kutta method is A-stable if its stability function is A-acceptable. The following results on A-acceptability in the case of Padé approximations holds. The interested reader can find a proof in [25].

Theorem 6.3 *All Padé approximations* $R_n^n(x)$ *to the exponential are A-acceptable.*

A consequence of this results is given by the following theorem, on the A-stability of all Gaussian RK methods, presented in Sect. 4.4.1.

Corollary 6.1 *All Gauss RK methods (see Sect. 4.4.1) are A-stable.*

Proof Gaussian RK methods depending on s stages have order $2s$, hence their stability function $R(\widehat{h})$ is an approximation of order $p = 2s$ to the exponential. The representation of the stability function provided by (6.13) is given by a rational function where both the numerator and the denominator have the same order s. Hence, $R(\widehat{h})$ is a rational approximation to the exponential of maximal order $2s$, i.e., it is the Padé approximation $R_s^s(\widehat{h})$, which is A-acceptable according to Theorem 6.3. Correspondingly, all Gaussian methods are A-stable. □

In other terms, when a Gaussian RK method is applied, there are no restrictions on the stepsize due to stability. As we will see in next chapter, this property is particularly relevant for stiff problems.

6.6.3 L-Stability

We conclude this section with a stability concept stronger than A-stability, which will be particularly useful in the integration of stiff problems, presented in Chap. 7. Let us first give this definition for Runge-Kutta methods.

Definition 6.17 An A-stable Runge-Kutta method (4.8), with stability function $R(z)$ given by (6.9), is *L-stable* if

$$\lim_{|z|\to\infty} R(z) = 0.$$

This concept, introduced by Ehle in [156], requires that the stability function of a Runge-Kutta method tends to zero when its argument tends to infinity. Let us provide few examples of L-stable methods.

Example 6.11 The stability function of the implicit Euler method (2.32) is given by

$$R(z) = \frac{1}{1-z}.$$

(continued)

Example 6.11 (continued)
Hence, since the method is A-stable, it is also L-stable. The stability function of the Radau method (4.26) is given by

$$R(z) = \frac{2(z+3)}{z^2 - 4z + 6}.$$

Also in this case, since the method is A-stable, it is also L-stable.

Clearly, according to above definition, all L-stable methods are also A-stable, A(α)-stable and A$_0$-stable. We now aim to provide a way to check if an A-stable Runge-Kutta method is also L-stable, according to a result given in [195].

Theorem 6.4 *An A-stable Runge-Kutta method* (4.8) *with nonsingular coefficient matrix A is L-stable if the last row of the matrix A is equal to* $b^\intercal$*, i.e.*

$$A^\intercal e_s = b,$$

where $e_s = [0 \;\; 0 \;\; \dots \;\; 1] \in \mathbb{R}^s$.

Proof We first compute the limit

$$\lim_{|z|\to\infty} R(z) = 1 - b^\intercal A^{-1} e. \tag{6.20}$$

Since $b^\intercal = e_s^\intercal A$, we have

$$\lim_{|z|\to\infty} R(z) = 1 - e_s^\intercal e = 0$$

and the thesis holds true. □

According to this result, all Radau IIA methods are L-stable, since the last row of A is equal to $b^\intercal$. Another similar condition of L-stability is given as follows.

Theorem 6.5 *An A-stable Runge-Kutta method (4.8) with nonsingular coefficient matrix A is L-stable if all the elements of the first column of A are equal to b_1, i.e.*

$$Ae_1 = b_1 e,$$

where $e_1 = [1 \;\; 0 \;\; \dots \;\; 0] \in \mathbb{R}^s$.

Proof Since, from the hypothesis,

$$e = \frac{1}{b_1} Ae_1,$$

from (6.20) we obtain that

$$\lim_{|z|\to\infty} R(z) = 1 - \frac{1}{b_1} b^\intercal A^{-1} Ae_1 = 0,$$

leading to the thesis. □

According to this result, all Radau IA methods are L-stable, since the first column of A is equal to b_1. We finally give the definition of L-stability in the general setting of multivalue methods (5.4).

Definition 6.18 A multivalue method (5.4) is *L-stable* if it is A-stable and

$$\lim_{|z|\to\infty} \rho(S(z)) = 0,$$

where ρ is the spectral radius and $S(z)$ is the stability matrix (6.14).

6.7 Order Stars

In developing numerical methods for ODEs (1.1) it is important to assess a good balance between order and stability properties. As expectable, for fixed values of the number s of internal stages, we cannot construct RK methods with unbounded stability region of any order; similarly, we cannot expect to develop linear multistep methods (3.1) of any order and with unbounded stability region for any fixed number k of steps. In this section we aim to clarify which is maximum attainable order for

A-stable linear multistep and RK methods: in the literature, results providing the relationships between order and stability properties are known as *order and stability barriers*.

Let us start with the case of Runge-Kutta methods. A relevant tool in developing order and stability barriers has been introduced in 1978 by Hairer, Nørsett and Wanner in their paper [342] and it is known in the literature as *order stars*. In order to present the theory of order stars, we introduce the following definition.

Definition 6.19 The *relative stability function* $\widetilde{R}(\widehat{h})$ associated to a Runge-Kutta method (4.8) with stability function $R(\widehat{h})$ is given by

$$\widetilde{R}(\widehat{h}) = \mathrm{e}^{-\widehat{h}} R(\widehat{h}). \tag{6.21}$$

The relative stability function is then given by ratio between the stability function and the exponential function, i.e., between a rational function and the function approximated by it. We observe that the stability function $R(\widehat{h})$ and the relative stability function $\widetilde{R}(\widehat{h})$ share the same poles and $|R(iy)| = |\widetilde{R}(iy)|$, $y \in \mathbb{R}$.

Definition 6.20 The *order star* S is the set of points in the complex plane such that $|\widetilde{R}(\widehat{h})| > 1$, where $\widetilde{R}(\widehat{h})$ is the relative stability function (6.21).

The following result holds true.

Lemma 6.1 *The stability function $R(\widehat{h})$ of a Runge-Kutta method (4.8) is A-acceptable if and only if the order star S has no intersection with the imaginary axis and $R(\widehat{h})$ has no poles in the negative half-plane.*

Proof The if part follows from the fact that, on the imaginary axis, $|\mathrm{e}^{\widehat{h}}| = 1$ and from the application of the maximum principle. The only if part follows from the definition of A-acceptability and order star. □

Example 6.12 Let us analyze the order star associated to the rational function

$$R(z) = \frac{1 + \frac{1}{3}z}{1 - \frac{2}{3}z + \frac{1}{6}z^2}, \tag{6.22}$$

that is the Padé approximation $R_2^1(x)$ to the exponential function. The corresponding order star is shown in Fig. 6.10, shaded in gray. According to Theorem 6.1, $R(z)$ is A-acceptable, since the order star has no intersection with the imaginary axis and the poles, displayed in the figure as empty circles, lie in the positive half-plane. Hence, $R(z)$ is the stability function of an A-stable method.

Figure 6.10 also shows a characteristic property of an order star, stated in the following proposition. A detailed proof can be found in [242, 342].

Lemma 6.2 *The boundary of the order star contains exactly two branches that tend to infinity.*

Order stars consist in the union of regions, called *fingers*, which can be bounded or unbounded. For instance, the order star in Fig. 6.10 contains two bounded fingers

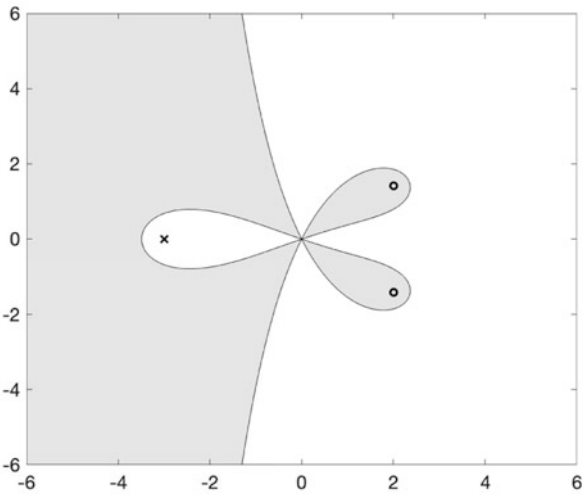


Fig. 6.10 Order stars associated to the rational function (6.22)

and an unbounded one. The complement of an order star instead consists in the union of *dual fingers*, which can be bounded or unbounded. Figure 6.10 contains a bounded dual finger and an unbounded one. A finger belonging to n sectors of B is called a *finger of multiplicity n*. As regards Fig. 6.10, each finger has multiplicity one. The following result clarifies the role of fingers and dual fingers of an order star, whose proof can be found in [224, 242, 342].

Lemma 6.3 *For a given order star, a bounded finger of multiplicity n contains at least n poles of the stability function $R(z)$, while each bounded dual finger of multiplicity n contains at least n zeros of the stability function $R(z)$.*

Figure 6.10 shows a pole in each bounded finger (displayed as empty circles) and a zero in the dual bounded finger (displayed as a cross). We now present a further lemma useful to prove the main results of this section. The interested reader can find its proof again in [224, 242, 342].

Lemma 6.4 *A function $R(z)$ is a rational approximation of order p to the exponential if and only if, in a neighborhood of the origin, its order star consists in $p+1$ sectors of angle $\pi/(p+1)$ separated by $p+1$ sectors of its dual with the same angle.*

We can now prove the main result of this section, known in the literature as *Ehle barrier*.

Theorem 6.6 *A Padé approximation $R_m^n(z)$ is A-acceptable if and only if $m-2 \leq n \leq m$.*

Proof Consider an A-acceptable approximation $R(z)$ of order p. Then, according to Lemma 6.4, there exist at least $[(p+1)/2]$ fingers starting in the left half-plane. Moreover, in force of Lemma 6.1, such fingers do not intersect with the imaginary axis and, due to Lemma 6.2, none of them is bounded. As a consequence, such $[(p+1)/2]$ fingers cluster in an unbounded multiple finger and $[(p+1)/2]-1$ bounded dual fingers in the left half-plane also exist. Due to Lemma 6.3, each of these dual fingers contain at least a zero of $R(z)$, hence $R(z)$ results to have at least

$[(p+1)/2]-1$ zeros. If $R(z)$ is the Padé approximation $R_m^n(z)$, then $p=n+m$ and $R_m^n(z)$ has n zeros. Then,

$$\left[\frac{p+1}{2}\right]-1 \leq n,$$

or, equivalently,

$$2n+2 \geq 2\left[\frac{p+1}{2}\right].$$

Moreover, $2[(p+1)/2] \geq p$, since $2[(p+1)/2]$ is equal to $p+1$ if p is odd and to p if p is even. Hence, we obtain $2n+2 \geq p$, i.e.,

$$n \geq m-2.$$

We leave to the reader the proof that we need $n \leq m$ in order to have A-acceptability. □

A consequence of this results is the order and stability barrier for Runge-Kutta methods, first stated by Daniel and Moore [139]. Formerly known as Daniel-Moore conjecture, it was proved in [342] with order stars theory (also see [224]).

Corollary 6.2 *The maximum attainable order of an A-stable Runge-Kutta method* (4.8) *is* $2s$*, where* s *is the number of stages.*

Proof The A-acceptability of a Padé approximation $R_m^n(z)$ requires $m-2 \leq n \leq m$, according to Theorem 6.6. Since, for a s-stage RK method, $m \leq s$, we have

$$p=n+m \leq 2m \leq 2s,$$

leading to the thesis. □

In other terms, we have proved that Gaussian RK methods are A-stable methods of maximal order. We conclude this section stating an analogous order and stability barrier for linear multistep methods (3.1), well-known in the literature as *second Dahlquist barrier*. The complete proof can be found, for instance, in [108, 195, 224].

Theorem 6.7

- *An explicit linear multistep method cannot be A-stable;*
- *the maximum attainable order for an A-stable linear multistep method is 2;*
- *the second order A-stable linear multistep method with smallest error constant is the trapezoidal method* (2.33).

In summary, we can conclude that Runge-Kutta methods allow to achieve a better compromise between order and stability, with respect to linear multistep methods, since Daniel-Moore barrier is less restrictive than the second Dahlquist barrier.

6.8 Exercises

1. Using Program 6.1, depict the boundary locus of Milne-Simpson method (3.2) and comment the results. Certainly, you can expect very poor stability properties, also taking into account the arguments provided in Example 6.3. It is also worth observing that, due to first Dahlquist barrier (see Theorem 3.5), Milne-Simpson method is a maximal order method, since it is a two-step method of order 4. To some extent, all degrees of freedom (given by its coefficients) have been employed to maximize the order of convergence, rather than its stability region.
2. Write a code in your favorite programming language that draws the stability region of a given RK method (4.8), by displaying its boundary locus. The program must take in input the coefficient matrix A and the vector of the weights b, shades the corresponding stability region and depicts its boundary.
3. Write a code in your favorite programming language that draws the stability region of a given multivalue method (5.4), by displaying its boundary locus. The program must take in input the coefficient matrices A, U, B, V, shades the corresponding stability region and depicts its boundary.
4. Construct the so-called *Padé table* up to 4, i.e., the table of Padé approximations $R^i_j(z)$, for i, j=1, 2, 3, 4, and discuss the A-acceptability of each element of the table.
5. Analyze the linear stability of all Lobatto methods introduced in Sect. 4.4.3.

6. Plot the boundary locus of the following LMMs and comment the results:

$$y_{n+2} - \frac{4}{3}y_{n+1} + \frac{1}{3}y_n = \frac{2}{3}hf_{n+2},$$

$$y_{n+3} - \frac{18}{11}y_{n+2} + \frac{9}{11}y_{n+1} - \frac{2}{11}y_n = \frac{6}{11}hf_{n+3},$$

$$y_{n+4} - \frac{48}{25}y_{n+3} + \frac{36}{25}y_{n+2} - \frac{16}{25}y_{n+1} + \frac{3}{25}y_n = \frac{12}{25}hf_{n+4}.$$

What happens to the stability regions when the order of the underlying difference equation increases? Why?

7. Design a linear stability theory for LMMs (3.1) and RK methods (4.8) in correspondence of a linear scalar test equation with a forcing term in the right-hand side, i.e.,

$$y'(t) = \lambda y(t) + a(t), \quad \lambda \in \mathbb{C}, \quad \mathrm{Re}(\lambda) < 0,$$

with $a : [t_0, +\infty) \to \mathbb{R}$.

8. Given the following second order ODE

$$y''(t) = -200y(t), \quad t \geq 0,$$

with $y(0) = 0$, $y'(0) = -20$, provide the stepsize restrictions needed to approximate its solution by the explicit Euler method (2.19). Give an experimental confirmation of the sharpness of the obtained bound, by employing Program 2.1.

9. Analyze the linear stability properties of the family of θ-methods

$$y_{n+1} = y_n + \theta hf_n + (1-\theta)hf_{n+1}, \quad \theta \in [0, 1].$$

Are there values of θ ensuring that the corresponding methods are A-stable?

10. Prove that the so-called TR-BDF2 method [251]

$$y^* = y_n + \frac{h}{4}\left(f_n + f\left(t_n + \frac{h}{2}, y^*\right)\right),$$

$$y_{n+1} = \frac{1}{3}\left(4y^* - y_n + hf_{n+1}\right),$$

is L-stable. Note that the formula for the computation of y^* is the trapezoidal method performing a step of length $h/2$, starting from y_n.

Chapter 7
Stiff Problems

Nevertheless, even though stiffness is phenomenologically well understood, the lack of a proper definition is unsatisfactory, not least from a pedagogical perspective. There is a need to define stiffness in a reasonably rigorous, simple and mathematically appealing way, rather than relying on descriptive approaches, in terms of operational criteria, method classes, software performance, or various notions of how "computationally demanding" a problem is or might be.

(Gustaf Söderlind, Laurent Jay, Manuel Calvo [326])

So far we have analyzed characteristic features of numerical methods for ODEs, mainly dealing with their accuracy and stability, with a focus on the approximation of a general Hadamard well-posed problem (1.1). However, in some cases, the choice of the numerical method to be used is driven by some features of the problems itself, which have to be properly taken into account. For instance, this is the case of the so-called *stiff* problems, usually occurring in mathematical modeling for several applications. This chapter is focused on the analysis of the main features of stiff problems and their numerical discretization; an exhaustive monograph on the topic is certainly given, for instance, by [195].

7.1 Looking for a Definition

"Stiff equations are multiscale problems". This sentence, contained in the first pages of the paper [79] by J. R. Cash (1947-2020) provides an important example of stiff equations, often occurring in the description of coupled physical systems having components which vary on very different time-scales: several examples which elaborate on this intuition can be found in [195, 328] and references therein. This situation is really very common in mathematical modeling: for instance, solving time-dependent partial differential equations by finite elements or finite differences

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_7

for the spatial discretization generally leads to stiff systems of ordinary differential equations, due to their intrinsic multiscale nature.

When coupling together deterministic models of processes that occur on different scales (also belonging to different physical systems), it looks more accurate to regard the whole system as a single model rather than the combination of simpler constituents. When models are coupled together in this way, obviously the number of involved variables becomes very large, as well as the range of all scales also increase. Equations representing such multiscale processes are thus particularly stiff (see [328]). A relevant multiscale problem in Life Science is the simulation of a beating heart (see, for instance, [279]). Within the heart, several coupled physical processes occur at each level and there are complex feedback mechanisms and processes occurring on multiple time scales. Multiscale models are extensively developed, for instance, also in immunological modelling to support the major challenge of identifying drug targets that efficiently interfere with viral replication in case of influenza [203]. Multiscale modeling provides an ideal framework to combine several aspects such as immune response, pharmacokinetics and comprehensive information on virus-host interactions as diverse cellular processes which can be simulated individually and incorporated as separate modules into a unifying framework.

The seminal paper by Curtiss and Hirschfelder [106] introduced the concept of stiffness for a differential problem. Anyway, although several decades have passed since this contribution, a definition of stiffness is not yet given in a widely shared manner. A gifted contribution on the topic is the paper [326] by G. Söderlind, L. Jay and M. Calvo, where the authors highlight that the opinion arising from many contributions provided after [106] agree with the difficulty to provide a rigorous definition of stiffness, although a stiff character can be clearly recognized in practice (see, for instance, [79, 141, 159, 195, 214, 242, 314, 315]). Despite the lack of a clear definition, it is well understood that *"stiff equations are equations where certain implicit methods, in particular BDF, perform better, usually tremendously better, than explicit ones"* (see [195], p. 1) and we aim to explain the reason of such a behavior in the remainder of this chapter.

We start our analysis from the following sentence provided by Dekker and Verwer in [141]: *"the essence of stiffness is that the solution to be computed is slowly varying but that perturbations exist which are rapidly damped"*. In other terms, a way to detect the stiffness of a problem relies in distinguishing slowly varying components of a solution from rapidly varying ones. To do this, let us consider the following inhomogeneous linear system of ODEs

$$y'(t) = Ay(t) + \varphi(t), \tag{7.1}$$

where $A \in \mathbb{R}^{d\times d}$ has d distinct eigenvalues $\lambda_i \in \mathbb{C}$, whose corresponding eigenvectors are denoted by $v_i \in \mathbb{C}^d$, $i = 1, 2, \ldots, d$; finally, the function

$\varphi(t) : [t_0, +\infty) \to \mathbb{R}^d$ is assumed to be smooth enough. Then, the general solution to Eq. (7.1) is represented by

$$y(t) = \tau(t) + \sigma(t),$$

with

$$\tau(t) = \sum_{i=1}^{d} c_i e^{\lambda_i t} v_i,$$

being $c_i \in \mathbb{R}$ and $\sigma(t)$ is a particular solution to (7.1). If $\mathrm{Re}(\lambda_i) < 0$, $i = 1, 2, \ldots, d$, then the solution $y(t)$ asymptotically approaches $\sigma(t)$, as t grows to infinity; for this reason, we denote $\tau(t)$ as the *transient* term and $\sigma(t)$ as the *steady-state* term of the solution. Let us observe that eigenvalue $\overline{\lambda}$ such that $|\mathrm{Re}(\lambda_i)| \leq |\mathrm{Re}(\overline{\lambda})|$, $i = 1, 2, \ldots, d$, corresponds to the fastest transient term, while the eigenvalue $\underline{\lambda}$ such that $|\mathrm{Re}(\underline{\lambda})| \leq |\mathrm{Re}(\lambda_i)|$, $i = 1, 2, \ldots, d$, provides the slowest transient component.

Clearly, when the discrepancy between $|\mathrm{Re}(\overline{\lambda})|$ and $|\mathrm{Re}(\underline{\lambda})|$ is large, we need to integrate the system with an extremely small stepsize h in order to let $h\overline{\lambda}$ fit into the absolute stability region of the employed method. Such a restriction, at least for stability properties, is certainly not required when the system is integrated by an implicit method with unbounded stability region. The following definition provides a relevant quantity in this analysis, introduced by Lambert [243].

Definition 7.1 The *stiffness* ratio associated to (7.1) is given by

$$\frac{|\mathrm{Re}(\overline{\lambda})|}{|\mathrm{Re}(\underline{\lambda})|}, \tag{7.2}$$

where $\underline{\lambda}$ and $\overline{\lambda}$ are the eigenvalues of the matrix A in (7.1) such that

$$|\mathrm{Re}(\lambda_i)| \leq |\mathrm{Re}(\overline{\lambda})|, \quad |\mathrm{Re}(\underline{\lambda})| \leq |\mathrm{Re}(\lambda_i)|, \quad i = 1, 2, \ldots, d.$$

An alternative definition of stiffness ratio can also be found in [33], where a presentation in view of a rigorous definition of stiffness is also presented. A large stiffness ratio can be assumed, in some cases, as a stiffness indicator. However, as highlighted by Byrne and Hindmarsh [74], this is neither necessary nor sufficient condition for stiffness. Indeed, there are scalar problems which are stiff even if their stiffness ratio is equal to 1; the stiffness ratio (7.2) is not taking into account the direction of the integration (since it makes sense only for eigenvalues with negative real part); moreover, the stiffness ratio highlights a global property independent on

the time scale, although stiffness may also vary along the solution. We also observe that the stiffness ratio characterizes linear systems (7.1) and may not be adequate for nonlinear ones by using the eigenvalues of the Jacobian matrix, as observed by Artemiev and Averina [17].

An alternative to the stiffness ratio was proposed by G. Dahlquist in [109], where he detected that the vector field of stiff systems has a large Lipschitz constant L. As developed in Theorem 3.1, the stepsize restriction (3.7) needed to guarantee the convergence of fixed point iterations for implicit linear multistep method may be vary severe if L is large, since the stepsize h behaves like $\frac{1}{L}$. In other terms, the number of points required to integrate (1.1) in $[t_0, T]$ would behave like $L(T - t_0)$ and, as a consequence, it may be vary large for stiff problems, also in relation to the length of the time window. Although defining stiffness through the Lipschitz constant of the vector field may fill some of the gaps of the stiffness ratio (e.g., it covers nonlinear problems and it is also related to time scales), it is still not enough to provide an exhaustive definition of stiffness. Indeed, as the stiffness ratio, it does not distinguish between the solution of the problem forward in time or in reverse time. Although a large Lipschitz constant cannot be assumed as robust criterion for stiffness, it suggests to avoid employing fixed point iterations when handling stiff problems, since their convergence would require excessively small stepsizes: indeed, as we discuss in Sect. 7.6, Newton iterations are preferred.

As announced, although stiff problems are hard to define, the effects of stiffness are very clear and next sections highlight how to detect them, both a-priori and a-posteriori.

Example 7.1 (Stiffness Ratio and Fake News Dynamics) In Example 1.4, we have presented Eq. (1.3) as SIR model for the diffusion of fake information [137]. Let us linearize the vector field around the initial value

$$\begin{bmatrix} S_0 & I_0 & R_0 \end{bmatrix}^\intercal = \begin{bmatrix} S(0) & I(0) & R(0) \end{bmatrix}^\intercal,$$

leading to

$$\begin{aligned} S'(t) &= \beta S_0 I_0 - \beta I_0 S(t) - \beta S_0 I(t) + \text{higher order terms}, \\ I'(t) &= -\beta S_0 I_0 + \beta I_0 S(t) + (\beta S_0 - \alpha) I(t) + \text{higher order terms}, \\ R'(t) &= \alpha I(t), \end{aligned} \tag{7.3}$$

(continued)

Example 7.1 (continued)
Correspondingly, let us compute the Jacobian matrix of the linear part of the vector field in (7.3), i.e.,

$$J_{\alpha,\beta}(S_0, I_0) = \begin{bmatrix} -\beta I_0 & -\beta S_0 & 0 \\ \beta I_0 & \beta S_0 - \alpha & 0 \\ 0 & \alpha & 0 \end{bmatrix},$$

whose spectrum consists in a null eigenvalue and two real eigenvalues $\lambda^{\min}_{\alpha,\beta}(S_0, I_0)$ and $\lambda^{\max}_{\alpha,\beta}(S_0, I_0)$, with $|\lambda^{\min}_{\alpha,\beta}(S_0, I_0)| < |\lambda^{\max}_{\alpha,\beta}(S_0, I_0)|$. Correspondingly, the ratio

$$\sigma_{\alpha,\beta}(S_0, I_0) = \frac{|\lambda^{\max}_{\alpha,\beta}(S_0, I_0)|}{|\lambda^{\min}_{\alpha,\beta}(S_0, I_0)|}, \tag{7.4}$$

is the stiffness ratio of the linearized problem (7.3). As highlighted in [137], the higher this stiffness ratio, the faster the transit of fake news will be. Indeed, smaller values of the stiffness ratio correspond to a slower achievement of the maximum number of infected people and, consequently, to a slower dispersion of fake news. A numerical evidence of this issue is object of Exercise 10 at the end of this chapter.

7.2 Prothero-Robinson Analysis

As pointed out by several authors (see, for instance, [45, 141, 172, 318, 326]), a useful tool to understand the effects of stiffness on numerical discretizations is the so-called Prothero-Robinson analysis, i.e., the behavior of explicit and implicit methods applied to the following scalar test problem, well-known in the literature as *Prothero-Robinson problem* [291]

$$\begin{cases} y'(t) = \lambda(y(t) - g(t)) + g'(t), & t \geq t_0, \\ y(t_0) = y_0 \neq g(t_0), \end{cases} \tag{7.5}$$

where λ is a complex parameter with negative real part and such that $|\mathrm{Re}(\lambda)| \gg 1$ and $g : [t_0, \infty) \to \mathbb{R}$. The exact solution

$$y(t) = \mathrm{e}^{\lambda(t-t_0)}\,(y_0 - g(t_0)) + g(t)$$

contains a transient term, given by the exponential part, and a steady-state term corresponding to the function $g(t)$ that is also a particular solution to the differential equation in (7.5) when $y_0 = g(t_0)$.

Let us analyze the behavior of the explicit and implicit Euler methods (2.19) and (2.32) applied to (7.5). The application of the explicit Euler method (2.19) to (7.5) leads to

$$y_{n+1} = a y_n + \varphi_n, \tag{7.6}$$

where

$$a = 1 + h\lambda, \qquad \varphi_n = -h\left(\lambda g(t_n) - g'(t_n)\right).$$

Replacing the exact solution in Eq. (7.6) yields

$$y(t_{n+1}) = a y(t_n) + \varphi_n + \frac{h^2}{2} y''(t_n) + O(h^3). \tag{7.7}$$

We denote by

$$e_n = y(t_n) - y_n$$

the error at the point t_n of the discretization and obtain, through side-by-side subtraction of (7.6) and (7.7), i.e.,

$$e_{n+1} = a e_n + \frac{h^2}{2} y''(t_n) + O(h^3).$$

In other terms, the propagated error is damped for any value of the stepsize h such that

$$|1 + h\lambda| < 1,$$

i.e., for any $h\lambda$ belonging to the stability region of Euler method (2.19), developed in Example 6.1. Clearly, when $\mathrm{Re}(\lambda)$ tends to $-\infty$, the integration requires a truly severe restriction for the stepsize h in order to fulfill the stability requirement. Actually, the stepsize restriction may result severe even for moderately large values for $|\mathrm{Re}(\lambda)|$. As mentioned, $g(t)$ is a particular solution to the differential equation in (7.5) when $y_0 = g(t_0)$. A method of order p is able to exactly solve polynomial solutions of degree up to p; therefore, if $g(t)$ is locally approximated by a polynomial of degree p, the error $|g(t) - P(t)|$ remains small on intervals of a certain length, here denoted as H. Such a time scale for the steady-state term $g(t)$ may substantially differ from that of the exponential $\mathrm{e}^{\lambda(t-t_0)}$ and a stable integration via the Euler method (2.19) may require the employ of a stepsize $h \ll H$, without

taking into account if the transient has decayed or not. If $H\mathrm{Re}(\lambda) \ll -1$, the problem is stiff; otherwise when $|H\mathrm{Re}(\lambda)| \lessapprox 1$ the problem is non-stiff.

Let us now focus on the implicit Euler method (2.32) that, applied to (7.5), assumes the form

$$y_{n+1} = \frac{y_n + \varphi_{n+1}}{1 - h\lambda}. \tag{7.8}$$

Replacing the exact solution in Eq. (7.8) yields

$$y(t_{n+1}) = \frac{y_n + \varphi_{n+1}}{1 - h\lambda} - \frac{h^2}{2(1 - h\lambda)} y''(t_n) + O(h^3), \tag{7.9}$$

i.e., by subtracting (7.8) from (7.9)

$$e_{n+1} = \frac{e_n}{1 - h\lambda} - \frac{h^2}{2(1 - h\lambda)} y''(t_n) + O(h^3).$$

Therefore, the propagated error is damped for any value of the stepsize h such that

$$\frac{1}{|1 - h\lambda|} < 1,$$

which is a condition certainly satisfied for any λ having negative real part. Hence, no stepsize restrictions due to stability are required when the implicit Euler is employed.

As one can realize from this analysis, the opening sentence of the book by Hairer and Wanner [195] is fully confirmed: *"stiff equations are equations where certain implicit methods perform better, usually tremendously better, than explicit ones"* (this sentence is a quote from [106]). An analogous investigation can be provided for a nonlinear system of Prothero-Robinson equations: the interested reader can find a detailed analysis of this case in [326].

7.3 Order Reduction of Runge-Kutta Methods

Section 4.5 has been devoted to introducing Runge-Kutta methods based on the collocation principle. We have realized, for instance, that all Gaussian RK formulae are collocation methods of order $2s$, being s the number of stages. We now aim to prove, through the following result, that collocation methods have a quite poor uniform order, that is equal to the number of the involved internal stages.

Theorem 7.1 *For $t \in [t_n, t_{n+1}]$, suppose that $P_n(t)$ is a given collocation polynomial (4.30). Then, there exists a positive constant C such that*

$$\|P_n(t) - y(t)\|_\infty \leq Ch^{s+1},$$

i.e., P_n given by (4.30) is an approximation of uniform order s to the solution $y(t)$ of (1.1).

Proof Consider Lagrangian formulation of the first derivative of P_n given by (4.34), that can be written in equivalent way as

$$P_n'(t_n + \eta h) = \sum_{i=1}^{s} f(t_n + c_i h, P_n(t_n + c_i h))L_i(\eta). \tag{7.10}$$

The corresponding interpolation error $E_h(\eta)$ (see [170, 292]) can be bounded by

$$E_h(\eta) \leq \frac{h^s}{s!} \max_{t\in[t_n,t_{n+1}]} \|y^{(s+1)}(t)\|.$$

Since,

$$y'(t_n + \eta h) = \sum_{i=1}^{s} f(t_n + c_i h, y(t_n + c_i h))L_i(\eta) + E_h(\eta),$$

we have, by subtraction,

$$y'(t_n + \eta h) - P_n'(t_n + \eta h) = \sum_{i=1}^{s} \Delta_i L_i(\eta) + E_h(\eta),$$

where

$$\Delta_i = f(t_n + c_i h, y(t_n + c_i h)) - f(t_n + c_i h, P_n(t_n + c_i h)).$$

Side-by-side integration from 0 to η leads to

$$y(t_n + \eta h) - P_n(t_n + \eta h) = h\left(\sum_{i=1}^{s} \Delta_i \int_0^\eta L_i(\tau)\mathrm{d}\tau + \int_0^\eta E_h(\tau)\mathrm{d}\tau\right).$$

As a consequence,

$$\max_{t\in[t_n,t_{n+1}]} \|y(t) - P_n(t)\| \leq hL\Lambda_{s-1} \max_{t\in[t_n,t_{n+1}]} \|y(t) - P_n(t)\| + \frac{h^{s+1}}{s!} \max_{t\in[t_n,t_{n+1}]} \|y^{(s+1)}(t)\|,$$

begin Λ_{s-1} the Lebesgue constant associated to the interpolation polynomial (7.10), i.e.,

$$\Lambda_{s-1} = \left\| \sum_{i=1}^{s} |L_i(t)| \right\|_\infty .$$

Then, the thesis holds true with

$$C = \frac{\max_{t\in[t_n,t_{n+1}]} \|y^{(s+1)}(t)\|}{s!(1 - hL\Lambda_{s-1})},$$

for sufficiently small values of h. □

Therefore, even if the maximum attainable order of s-stage collocation methods is $2s$ (and it is obtained by using Gaussian collocation points), the uniform order is only s. In other terms, RK methods may exhibit effective order s of convergence, even if the theoretical order in the grid points is higher: this phenomenon, known as *order reduction*, is typical of RK methods, especially when they are applied to solve stiff problems. Prothero and Robinson [291] observed some order reduction effects for certain Runge-Kutta methods. A detailed analysis of order reduction phenomenon for RK methods applied to stiff problems has first been provided in [167], here suggested to the interested reader as a reference where proofs of order reduction for RK methods are presented, also outside collocation. We now give a numerical evidence of order reduction for Gaussian RK methods applied to stiff problems.

Example 7.2 We now provide a numerical experiment based on the application of the two-stage Runge-Kutta method on Gaussian points (4.25) to the Prothero-Robinson problem (7.5) in $[0,\ 100]$, with $g(t) = \sin(t)$, initial value $y_0 = 0$ and for $\lambda = -10^3, -10^5$. The exact solution is $y(t) = \sin(t)$.

Table 7.1 displays the errors at the endpoint of the integration interval, computed as the infinity norm of the difference between the exact solution and the numerical solution y_{RK} computed by (4.25). As visible from the Table, since for $\lambda = -10^3$ Prothero-Robinson problem (7.5) appears to be non-

(continued)

Example 7.2 (continued)
stiff, the experimental order of convergence is the expected one, i.e., $p = 4$. However, when $\lambda = -10^5$, the problem is stiff and the Gaussian RK method exhibits order reduction, converging with order $p = 2$, equal to the number of internal stages. We observe that all order estimates reported in Table 7.1 have been computed through formula (3.23).

7.4 Discretizations Free from Order Reduction

As explained in the previous section, Runge-Kutta methods are exposed to a severe order reduction when applied to stiff problems. In this section we present an alternative to Runge-Kutta methods, given by highly stable formulae based on a proper modification of the collocation technique presented in Sect. 4.5, that do not suffer from order reduction when applied to stiff problems. The development and analysis of this modified collocation technique is detailed object of [123–126, 131, 228] and relies on extending the collocation principle to the family of two-step Runge-Kutta methods (5.15) and multivalue methods (5.2).

7.4.1 Two-Step Collocation Methods

To provide a dense output version of TSRK methods, with reference to a single step from t_n to t_{n+1}, we compute a unique algebraic polynomial

$$P_n(t_n + \eta h) = \varphi_0(\eta) y_{n-1} + \varphi_1(\eta) y_n + h \sum_{j=1}^{s} \left(\chi_j(\eta) f_j^{[n-1]} + \psi_j(\eta) f_j^{[n]} \right), \tag{7.11}$$

Table 7.1 Example 7.2: error in the final integration point associated to the application of the two-stage Gaussian Runge-Kutta method (4.25) to (7.5) and order estimation. The employed stepsize is $100/2^k$, for various values of the integer k. Each column "error" reports the value of the deviation $\|y(100) - y_{\text{RK}}\|_\infty$

$\lambda = -10^3$			$\lambda = -10^5$		
k	Error	p	k	Error	p
11	$9.64 \cdot 10^{-6}$		7	$2.06 \cdot 10^{-2}$	
12	$6.86 \cdot 10^{-7}$	3.81	8	$3.65 \cdot 10^{-3}$	2.50
13	$4.40 \cdot 10^{-8}$	3.96	9	$6.87 \cdot 10^{-4}$	2.41
14	$2.76 \cdot 10^{-9}$	3.99	10	$1.39 \cdot 10^{-4}$	2.30
15	$1.73 \cdot 10^{-10}$	4.00	11	$3.06 \cdot 10^{-5}$	2.19

with $\eta \in [0, 1]$ and $f_j^{[n]} = t(t_n + c_j h, P_n(t_n + c_j h))$, given as linear combination of the following basis of algebraic polynomials

$$\{\varphi_0(\eta),\ \varphi_1(\eta),\ \chi_j(\eta),\ \psi_j(\eta),\ j = 1, 2, \ldots, s\}. \tag{7.12}$$

Once $P_n(t_n + \eta h)$ is computed, its evaluation for $\eta = 1$ gives the approximate solution at t_{n+1}, i.e., $y_{n+1} = P_n(t_n + h)$. The unknown basis functions (7.12) are recovered by assuming that (7.11) satisfies interpolation conditions on two adjacent grid points

$$P_n(t_{n-1}) = y_{n-1}, \quad P_n(t_n) = y_n \tag{7.13}$$

and collocation conditions on two adjacent intervals of the discretization

$$\begin{aligned} P_n'(t_{n-1} + c_i h) &= f(t_{n-1} + c_i h, P_n(t_{n-1} + c_i h)), \\ P_n'(t_n + c_i h) &= f(t_n + c_i h, P_n(t_n + c_i h)), \end{aligned} \tag{7.14}$$

$i = 1, 2, \ldots, s$. With a formalism similar to that introduced in Sect. 4.5, we denote $P(t_n + \eta h)$ as *two-step collocation polynomial*. The corresponding numerical scheme

$$y_{n+1} = P_n(t_n + h), \tag{7.15}$$

where P_n is the two-step collocation polynomial (7.11), is denoted as *two-step collocation method*.

The counterpart of the interpolation conditions (7.13) on the basis functions (7.12) is given by

$$\begin{aligned} &\varphi_0(-1) = 1,\ \varphi_1(-1) = 0,\ \chi_j(-1) = 0,\ \psi_j(-1) = 0, \\ &\varphi_0(0) = 0, \quad \varphi_1(0) = 1, \quad \chi_j(0) = 0, \quad \psi_j(0) = 0, \end{aligned} \tag{7.16}$$

while for the collocation conditions (7.14) we have

$$\begin{aligned} &\varphi_0'(c_i - 1) = 0,\ \varphi_1'(c_i - 1) = 0,\ \chi_j'(c_i - 1) = \delta_{ij},\ \psi_j'(c_i - 1) = 0, \\ &\varphi_0'(c_i) = 0, \qquad \varphi_1'(c_i) = 0, \qquad \chi_j'(c_i) = 0, \qquad \psi_j'(c_i) = \delta_{ij}, \end{aligned} \tag{7.17}$$

where δ_{ij} is the usual Kronecker delta, $i, j = 1, 2, \ldots, s$.

The basis functions (7.12) are determined in such a way that $P(t_n + \eta h)$ is an approximation to $y(t_n + \eta h)$, $\eta \in [0, 1]$, of uniform order p, i.e.,

$$\lim_{\substack{n \to \infty \\ nh = t - t_0}} P(t_n + \eta h) = y(t_n + \eta h), \quad \text{for any } \eta \in [0, 1].$$

In other terms, an approximant of uniform order p provides a discretization of the same order in any point of the integration interval, not only in the grid points. The analysis of uniform order relies on the following result.

Theorem 7.2 *Assuming that the vector field of the differential problem (1.1) is sufficiently smooth, then the two-step collocation method (7.15) has uniform order p if the following conditions are satisfied*

$$\begin{cases} \varphi_0(\eta) + \varphi_1(\eta) = 1, \\ \dfrac{(-1)^k}{k!}\varphi_0(\eta) + \displaystyle\sum_{j=1}^{s} \left(\chi_j(\eta) \frac{(c_j - 1)^{k-1}}{(k-1)!} + \psi_j(\eta) \frac{c_j^{k-1}}{(k-1)!} \right) = \frac{\eta^k}{k!}, \end{cases} \tag{7.18}$$

$\eta \in [0, 1]$, $k = 1, 2, \ldots, p$.

Proof We investigate the local discretization error $\xi(t_n + \eta h)$ associated to (7.11), i.e., the residuum obtained replacing $P(t_n + \eta h)$ by $y(t_n + \eta h)$, $P(t_n + c_j h)$ by $y(t_n + c_j h)$, $j = 1, 2, \ldots, s$, y_{n-1} by $y(t_{n-1})$ and y_n by $y(t_n)$ in (7.11), where $y(t)$ is the exact solution to (1.1). This leads to

$$\begin{aligned} \xi(t_n + \eta h) &= y(t_n + \eta h) - \varphi_0(\eta) y(t_n - h) - \varphi_1(\eta) y(t_n) \\ &\quad - h \sum_{j=1}^{s} \left(\chi_j(\eta) y'(t_n + (c_j - 1)h) + \psi_j(\eta) y'(t_n + c_j h) \right), \end{aligned} \tag{7.19}$$

$\eta \in [0, 1]$. We expand the right-hand side of (7.19) in Taylor series around t_n and collect in powers of h, obtaining

$$\begin{aligned} &\xi(t_n + \eta h) \\ &= \left(1 - \varphi_0(\eta) - \varphi_1(\eta)\right) y(t_n) + \sum_{k=1}^{p+1} \left(\frac{\eta^k}{k!} - \frac{(-1)^k}{k!} \varphi_0(\eta) \right) h^k y^{(k)}(t_n) \\ &\quad - \sum_{k=1}^{p+1} \sum_{j=1}^{m} \left(\chi_j(\eta) \frac{(c_j - 1)^{k-1}}{(k-1)!} + \psi_j(\eta) \frac{c_j^{k-1}}{(k-1)!} \right) h^k y^{(k)}(t_n) + O(h^{p+2}). \end{aligned}$$

Collecting in powers of h and equating to zero the coefficients of the powers of up to p leads to (7.18). □

As a consequence of Theorem 7.2, if the two-step collocation method (7.15) has uniform order p, then the local discretization error takes the form

$$\xi(t_n + \eta h) = h^{p+1} C_p(\eta) y^{(p+1)}(t_n) + O(h^{p+2}), \quad \eta \in [0, 1],$$

where the principal error function $C_p(s)$ is defined by

$$C_p(\eta) = \frac{\eta^{p+1}}{(p+1)!} - \frac{(-1)^{p+1}}{(p+1)!}\varphi_0(\eta) - \sum_{j=1}^{m} \left(\chi_j(\eta) \frac{(c_j - 1)^p}{p!} + \psi_j(\eta) \frac{c_j^p}{p!} \right).$$

The set of uniform order conditions (7.18) is a linear system of $p + 1$ equations in $2s + 2$ unknowns, i.e., the $2s + 2$ basis functions (7.12). As a consequence, in order to ensure the compatibility of system (7.18), p can be at most equal to $2s + 1$. Then, the following result holds true.

Corollary 7.1 *The maximum attainable uniform order for a two-step collocation method* (7.15) *is* $2s + 1$.

For a proof of the uniqueness of the solution to (7.18) when $p = 2s + 1$, the interested reader can refer to [115], where a proof of the equivalence between the numerical scheme (7.15) and TSRK method (5.15) with

$$\theta = \varphi_0(1), \quad v_j = \psi_j(1), \quad w_j = \chi_j(1), \quad u_i = \varphi_0(c_i), \quad a_{ij} = \psi_j(c_i), \quad b_{ij} = \chi_j(c_i),$$

$i, j = 1, 2, \ldots, s$, is also given. Another relevant property of the solutions to (7.18) is given in the following result.

Theorem 7.3 *The algebraic polynomials* (7.12) *obtained as solutions to* (7.18) *satisfy the interpolation and collocation conditions (7.16) and (7.17).*

Proof The interpolation conditions (7.16) follow immediately by replacing $\eta = 0$ and $\eta = -1$ in (7.18) for $p = 2s + 1$. In order to recover the collocation conditions (7.17), we differentiate each condition in (7.18) with respect to η, obtaining

$$\begin{cases} \varphi_0'(\eta) + \varphi_1'(\eta) = 0, \\ \dfrac{(-1)^k}{k!}\varphi_0'(\eta) + \displaystyle\sum_{j=1}^{s} \left(\chi_j'(\eta) \frac{(c_j - 1)^{k-1}}{(k-1)!} + \psi_j'(\eta) \frac{c_j^{k-1}}{(k-1)!} \right) = \frac{\eta^{k-1}}{(k-1)!}, \end{cases}$$

$k = 1, 2, \ldots, 2s + 1$, and replace $\eta = c_i$ and $\eta = c_i - 1$, $i = 1, 2, \ldots, s$. □

In summary, the basis functions computed by solving (7.18) with $p = 2s + 1$ automatically satisfy all interpolation conditions (7.16) and all collocation ones (7.17). As a consequence, the corresponding two-step collocation polynomial (7.11) satisfies (7.13) and (7.14).

7.4.2 Almost Collocation Methods

Two-step collocation methods of maximal order $p = 2s + 1$ are not suitable to approach stiff problems, since they violate the Daniel-Moore barrier proved by Corollary 6.2 and, as a consequence, they cannot be A-stable. Therefore, let us look for A-stable methods of uniform order $p = 2s$ by relaxing one of the order conditions in (7.18). In other terms, we solve the system of $p + 1$ uniform order conditions (7.18) up to $p = 2s$ and since the unknowns are $2s + 2$, one of them has to be fixed a-priori. The solvability of such a relaxed system is discussed in the following result.

Theorem 7.4 *Assuming that $c_i \neq c_j$, and $c_i \neq c_j - 1$ for $i \neq j$, the system (7.18) of continuous order conditions $p = s + r$, $r = 1, 2, \ldots, s$, has a unique solution $\varphi_1(\eta)$, $\chi_j(\eta)$, $j = s - r + 1, s - r + 2, \ldots, s$, and $\psi_j(\eta)$, $j = 1, 2, \ldots, s$, for any given $\varphi_0(\eta)$ and $\chi_j(\eta)$, $j = 1, 2, \ldots, s - r$.*

Proof Observe that the polynomial $\varphi_1(\eta)$ is uniquely determined from the first equation of (7.18). The proof follows from the fact that the matrices of these systems (7.18) corresponding to $\chi_j(\eta)$, $j = s - r + 1, s - r + 2, \ldots, s$, are Vandermonde matrices and, therefore, the solution exists and is unique. □

In particular, for the development of methods of uniform order $p = 2s$, we choose the algebraic polynomial $\varphi_0(\eta)$ of order at most $2s$, satisfying the interpolation and collocation conditions

$$\varphi_0(0) = 0, \quad \varphi_0'(c_i) = 0,$$

$i = 1, 2, \ldots, s$. As a consequence, $\varphi_0(\eta)$ factors out as

$$\varphi_0(\eta) = \eta\big(q_0 + q_1\eta + \cdots + q_{2s-1}\eta^{2s-1}\big), \tag{7.20}$$

with

$$q_0 + 2q_1c_i + \cdots + 2sq_{2s-1}c_i^{2s-1} = 0, \tag{7.21}$$

$i = 1, 2, \ldots, s$. Hence, $\varphi_0(\eta)$ does not fulfill all the interpolation conditions (7.16) and the collocation ones (7.17) occurring in the case of two-step collocation methods. We observe that all the other basis functions in (7.12) inherit the same conditions imposed on $\varphi_0(\eta)$ via order conditions; the interested reader can find a proof of this issue in [115, 131, 228].

As a consequence, the polynomial $P_n(t_n + \eta h)$ defined by (7.11) and arising as linear combination of $\varphi_0(\eta)$ and the remaining basis functions (7.12) computed by (7.18) with $p = 2s$, satisfies the interpolation and collocation conditions

$$P_n(t_n) = y_n, \quad P_n'(t_n + c_i h) = f_n(t_n + c_i h, P_n(t_n + c_i h)),$$

$i = 1, 2, \ldots, s$. However, in general, P_n does not satisfy the interpolation and collocation conditions

$$P_n(t_{n-1}) = y_{n-1}, \quad P_n'(t_{n-1} + c_i h) = f(t_{n-1} + c_i h, P_n(t_{n-1} + c_i h)),$$

$i = 1, 2, \ldots, s$. The corresponding method

$$y_{n+1} = P_n(t_{n+1})$$

is denoted in the literature as *two-step almost collocation method*.

For the development of A-stable two-step almost collocation methods, we now need to compute the corresponding stability matrix, arising from the application of (7.11) to the Dahlquist test problem (6.1). Assuming that

$$P_n(t_n + ch) = \begin{bmatrix} P_n(t_n + c_1 h) \\ P_n(t_n + c_2 h) \\ \vdots \\ P_n(t_n + c_s h) \end{bmatrix}, \quad \varphi_0(c) = \begin{bmatrix} \varphi_0(c_1) \\ \varphi_0(c_2) \\ \vdots \\ \varphi_0(c_s) \end{bmatrix}, \quad \varphi_1(c) = \begin{bmatrix} \varphi_1(c_1) \\ \varphi_1(c_2) \\ \vdots \\ \varphi_1(c_s) \end{bmatrix},$$

$$v^\intercal = \begin{bmatrix} \psi_1(1) & \psi_2(1) & \cdots & \psi_s(1) \end{bmatrix}^\intercal, \quad w^\intercal = \begin{bmatrix} \chi_1(1) & \chi_2(1) & \cdots & \chi_s(1) \end{bmatrix}^\intercal,$$

and

$$A = \left[\psi_j(c_i)\right]_{i,j=1}^s, \quad B = \left[\chi_j(c_i)\right]_{i,j=1}^s,$$

we obtain

$$P_n(t_n + ch) = \varphi_0(c) y_{n-1} + \varphi_1(c) y_n + \widehat{h}\,(B P_n(t_{n-1} + ch) + A P_n(t_n + ch)),$$

$$y_{n+1} = \varphi_0(1) y_{n-1} + \varphi_1(1) y_n + \widehat{h}\left(w^\intercal P_n(t_{n-1} + ch) + v^\intercal P_n(t_n + ch)\right),$$

with $\widehat{h} = h\lambda$. Hence, the stage values satisfy the relation

$$P(t_n + ch) = \Lambda \left(\varphi_0(c) y_{n-1} + \varphi_1(c) y_n + \widehat{h} B P(t_{n-1} + ch)\right), \tag{7.22}$$

where $\Lambda = \left(I - \widehat{h}A\right)^{-1}$. Finally,

$$\begin{aligned} y_{n+1} = &\left(\varphi_0(1) + \widehat{h} v^{\intercal} \Lambda \varphi_0(c)\right) y_{n-1} + \left(\varphi_1(1) + \widehat{h} v^{\intercal} \Lambda \varphi_1(c)\right) y_n \\ &+ \widehat{h} \left(w^{\intercal} + \widehat{h} v^{\intercal} \Lambda B\right) P_n(t_{n-1} + ch). \end{aligned} \tag{7.23}$$

Equations (7.22) and (7.23) are then equivalent to the following recurrence relation

$$\begin{bmatrix} y_{n+1} \\ y_n \\ P_n(t_n + ch) \end{bmatrix} = M(\widehat{h}) \begin{bmatrix} y_n \\ y_{n-1} \\ P_n(t_{n-1} + ch) \end{bmatrix},$$

where the $(m+2) \times (m+2)$ matrix

$$M(\widehat{h}) = \begin{bmatrix} \varphi_1(1) + \widehat{h} v^{\intercal} \Lambda \varphi_1(c) & \varphi_0(1) + \widehat{h} v^{\intercal} \Lambda \varphi_0(c) & \widehat{h} \left(w^{\intercal} + \widehat{h} v^{\intercal} \Lambda B\right) \\ 1 & 0 & 0 \\ \Lambda \varphi_1(c) & \Lambda \varphi_0(c) & \widehat{h} \Lambda B \end{bmatrix},$$

is the stability matrix of the two-step collocation method (7.11). The almost collocation method has the same stability matrix, since it inherits the same form for the collocation polynomial, as in the full two-step case. The characteristic polynomial

$$p(\omega, \widehat{h}) = \det\left(\omega I - M(\widehat{h})\right).$$

is the stability function of (7.11).

Let us now provide an example of two-step almost collocation method, depending on one stage. Further examples can be found in [123–125, 131, 228].

Example 7.3 We aim to construct a two-step almost collocation method depending on one internal stage, of order 2 and A-stable. We assume that $\varphi_0(\eta)$ satisfies (7.20) and (7.21). As a consequence, it assumes the form

$$\varphi_0(s) = q_0 s \left(1 - \frac{1}{2c} s\right),$$

(continued)

Example 7.3 (continued)
with $q_0 \in \mathbb{R}$. Solving the linear system of order conditions (7.18) for $s = 1$ and $p = 2$ leads to

$$\varphi_1(s) = 1 - q_0 s + \frac{q_0}{2c} s^2, \quad \chi(s) = \frac{s}{4c}(2c - s)(2c + q_0(1 + 2c)),$$
$$\psi(s) = \frac{s^2}{2} - s(c - 1) + q_0 s \left(c - \frac{1}{2}\right)\left(\frac{s}{2c} - 1\right).$$

Above basis functions recover a two-parameter family of one-stage and second order two-step almost collocation methods, depending on the real numbers q_0 and c. The values of q_0 and c for which the corresponding method is also A-stable have computed in [131], by means of the so-called Schur criterion [242, 312]. Choosing, for instance, $q_0 = -1$ and $c = 3/4$ leads to the second order A-stable two-step almost collocation method (7.11) depending on the basis functions

$$\varphi_0(\eta) = \left(\frac{2}{3}\eta - 1\right)\eta, \quad \varphi_1(\eta) = 1 - \varphi_0(\eta),$$
$$\chi(\eta) = \frac{1}{2}\varphi_0(\eta), \qquad \psi(\eta) = \left(\frac{1}{2} + \frac{1}{3}\eta\right)\eta,$$

and the corresponding stability polynomial is given by

$$p(\omega, \widehat{h}) = \omega\left(\left(3 - \frac{27}{16}\widehat{h}\right)\omega^2 - \left(4 + \frac{5}{8}\widehat{h}\right)\omega + \left(1 + \frac{5}{16}\widehat{h}\right)\right).$$

Example 7.4 As shown in Example 7.2, the two-stage Runge-Kutta method on Gaussian points (4.25) exhibits order reduction in solving Prothero-Robinson problem (7.5) for $\lambda = -10^5$. We now provide a numerical evidence based on the application of the two-step almost collocation method developed in Example 7.3 to the same problem.

Table 7.2 displays the errors in the endpoint of the integration interval, computed as the infinity norm of the difference between the exact solution minus the numerical solution y_{TSAC} computed by the two-step almost collocation method developed in Example 7.3. This method does not exhibit order reduction, even when the problem is stiff. We observe that, when $|\lambda|$ is large enough, the effects of the leading error term become negligible and the method converges with order $p = 3$. Clearly, when λ is large enough, the experimental order is the theoretical one ($p = 2$) as visible, for instance, in the column for $\lambda = -1$.

Table 7.2 Example 7.4: error in the final integration point associated to the application of the two-step almost collocation method developed in Example 7.3 to (7.5) and order estimation. The employed stepsize is $100/2^k$, for various values of the integer k. Each column "error" reports the value of $\|y(100) - y_{\text{TSAC}}\|_\infty$

$\lambda = -1$			$\lambda = -10^3$			$\lambda = -10^5$		
k	Error	p	k	Error	p	k	Error	p
11	$5.99 \cdot 10^{-5}$		7	$1.43 \cdot 10^{-2}$		7	$1.44 \cdot 10^{-2}$	
12	$1.69 \cdot 10^{-5}$	1.83	8	$2.95 \cdot 10^{-3}$	2.27	8	$2.97 \cdot 10^{-3}$	2.27
13	$4.46 \cdot 10^{-6}$	1.92	9	$4.38 \cdot 10^{-4}$	2.75	9	$4.43 \cdot 10^{-4}$	2.74
14	$1.14 \cdot 10^{-6}$	1.96	10	$5.79 \cdot 10^{-5}$	2.92	10	$5.94 \cdot 10^{-5}$	2.90
15	$2.90 \cdot 10^{-7}$	1.98	11	$7.24 \cdot 10^{-6}$	2.99	11	$7.63 \cdot 10^{-6}$	2.96

7.4.3 *Multivalue Collocation Methods Free from Order Reduction*

Multivalue collocation methods introduced in Sect. 5.5 (also see [126]) are also free from order reduction when applied to stiff systems since, according to Theorem 5.4, their order of convergence is uniform overall the entire integration interval. Let us provide an example, confirming this theoretical expectation.

Example 7.5 Let us provide a numerical test based on the application of the second order A-stable multivalue numerical method in Example 5.4 for $c = \frac{3}{2}$, i.e.,

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{c|cc} \frac{3}{4} & 1 & \frac{3}{4} \\ \hline \frac{1}{3} & 1 & \frac{2}{3} \\ \frac{2}{3} & 0 & \frac{1}{3} \end{array}\right]. \tag{7.24}$$

to the Prothero-Robinson problem (7.5) in $[0,\ 10]$, with $g(t) = \sin(t)$, initial value $y_0 = 0$ and for various values of λ. As visible in Tables 7.3 and 7.4, for $\lambda = -10^3$, the problem is not so stiff and we can see order of convergence $p = 2$ for both the Gaussian RK method (4.23) and multivalue method (7.24). However, when $\lambda = -10^6$, the problem is stiff and the Runge-Kutta method exhibits the order reduction phenomenon and its order of convergence drops to about $p = 1$, while this is not the case for the multivalue collocation methods (7.24).

Table 7.3 Example 7.5: observed errors (in the final step point) and orders of convergence for the one-stage Gaussian method (4.23) applied to the Prothero-Robinson problem

h	Error ($\lambda = -10^3$)	p	Error ($\lambda = -10^6$)	p
1/10	$6.80 \cdot 10^{-4}$		$6.81 \cdot 10^{-4}$	
1/20	$1.70 \cdot 10^{-4}$	2.00	$3.24 \cdot 10^{-4}$	1.07
1/40	$4.25 \cdot 10^{-5}$	2.00	$1.58 \cdot 10^{-4}$	1.04
1/80	$1.06 \cdot 10^{-5}$	2.00	$7.83 \cdot 10^{-5}$	1.01

Table 7.4 Example 7.5: observed errors (in the final step point) and orders of convergence for the multivalue method (7.24) applied to the Prothero-Robinson problem

h	Error ($\lambda = -10^3$)	p	Error ($\lambda = -10^6$)	p
1/10	$7.16 \cdot 10^{-7}$		$1.53 \cdot 10^{-9}$	
1/20	$1.75 \cdot 10^{-7}$	2.03	$3.81 \cdot 10^{-10}$	2.01
1/40	$4.37 \cdot 10^{-8}$	2.00	$9.19 \cdot 10^{-11}$	2.05
1/80	$1.09 \cdot 10^{-9}$	2.00	$2.11 \cdot 10^{-11}$	2.12

7.5 Stiffly-Stable Methods: Backward Differentiation Formulae

We conclude this chapter by presenting a relevant family of linear multistep methods particularly suited for the numerical solution of stiff systems, well-known in the literature as *backward differentiation formulae* (BDF). This is a family of implicit k-step methods of the form

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k}, \tag{7.25}$$

whose right-hand side consists in a single function evaluation related to the point t_{n+k}. These methods have been introduced by Curtiss and Hirschfelder in [106], specifically for the integration of stiff problems. BDF methods are developed via polynomial interpolation directly applied to the differential problem (1.1) and not to its integral formulation, as it happens for Adams methods.

In correspondence to the set of distinct $k + 1$ points

$$(t_n, y_n), \quad (t_{n+1}, y_{n+1}), \quad \ldots, \quad (t_{n+k}, y_{n+k}),$$

we develop a unique interpolation polynomial of degree k, here denoted as $P_k(t)$. Then $P_k(t)$ is an approximation to $y(t)$ in the interval $[t_n, t_{n+k}]$; as a consequence,

$$P_k'(t_{n+k}) \approx f(t_{n+k}, y(t_{n+k})).$$

Example 7.6 Let us compute the one-step BDF method by polynomial interpolation with respect to the set of nodes

$$(t_n, y_n), \quad (t_{n+1}, y_{n+1}).$$

The interpolation polynomial $P_1(t)$ is given by

$$P_1(t) = \frac{t - t_{n+1}}{t_n - t_{n+1}} y_n + \frac{t - t_n}{t_{n+1} - t_n} y_{n+1}.$$

Then,

$$P_1'(t) = -\frac{y_n}{h} + \frac{y_{n+1}}{h}.$$

As a consequence, we obtain the method

$$y_{n+1} - y_n = h f_{n+1},$$

that is the implicit Euler method (2.32).

Example 7.7 We now compute the two-step BDF method by polynomial interpolation with respect to the nodes

$$(t_n, y_n), \quad (t_{n+1}, y_{n+1}), \quad (t_{n+2}, y_{n+2}).$$

The interpolation polynomial $P_2(t)$ is then given by

$$P_2(t) = \frac{t - t_{n+1}}{t_n - t_{n+1}} \frac{t - t_{n+2}}{t_n - t_{n+2}} y_n + \frac{t - t_n}{t_{n+1} - t_n} \frac{t - t_{n+2}}{t_{n+1} - t_{n+2}} y_{n+1} + \frac{t - t_n}{t_{n+2} - t_n} \frac{t - t_{n+1}}{t_{n+2} - t_{n+1}} y_{n+2},$$

i.e.,

$$P_2(t) = \frac{(t - t_{n+1})(t - t_{n+2})}{2h^2} y_n - \frac{(t - t_n)(t - t_{n+2})}{h^2} y_{n+1} + \frac{(t - t_n)(t - t_{n+1})}{2h^2} y_{n+2}.$$

(continued)

Example 7.7 (continued)
Then,

$$P_2'(t) = \frac{2t - t_{n+1} - t_{n+2}}{2h^2} y_n - \frac{2t - t_n - t_{n+2}}{h^2} y_{n+1} + \frac{2t - t_n - t_{n+1}}{2h^2} y_{n+2}.$$

As a consequence, we obtain the second order method

$$y_{n+2} - \frac{4}{3} y_{n+1} + \frac{1}{3} y_n = \frac{2}{3} h f_{n+2}.$$

Table 7.5 shows the coefficients of BDF methods up to $k = 6$, which are convergent methods of order k. BDF methods with $k \geq 7$ are not zero-stable, as proved by Cryer in [104].

Let us now focus on their linear stability properties. To this purpose, since BDF methods fall in the family of linear multistep methods (3.1), we analyze their boundary loci (6.17) using the coding reported in Program 6.1.

The results are depicted in Fig. 7.1. The methods for $k = 1$ and $k = 2$ are both A-stable while, in all the other cases, the negative real axis is always contained in each stability region. The peculiar shapes of the stability regions make BDF methods particularly suitable for stiff problems, in particular when the eigenvalues producing the fastest transients are located to the left of $\mathrm{Re}(\widehat{h}) = -a$, with $a > 0$, and all the other ones are close to the origin with a small imaginary part. Such a peculiar shape for the stability region led to the following definition, provided by Gear in [171].

Definition 7.2 A numerical method for (1.1) is *stiffly-stable* if its stability region contains the set

$$\mathcal{S} = \{\widehat{h} : \mathrm{Re}(\widehat{h}) < -a\} \cup \{\widehat{h} : -a \leq \mathrm{Re}(\widehat{h}) < 0, \ -b \leq \mathrm{Im}(\widehat{h}) \leq b\},$$

for given values of $a > 0$ and $b > 0$.

Table 7.5 Coefficients of the BDF methods (7.25) up to $k = 6$

k	α_6	α_5	α_4	α_3	α_2	α_1	α_0	β_k
1						1	-1	1
2					1	$-\frac{4}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
3				1	$-\frac{18}{11}$	$\frac{9}{11}$	$-\frac{2}{11}$	$\frac{6}{11}$
4			1	$-\frac{48}{25}$	$\frac{36}{25}$	$-\frac{16}{25}$	$\frac{3}{25}$	$\frac{12}{25}$
5		1	$-\frac{300}{137}$	$\frac{300}{137}$	$-\frac{200}{137}$	$\frac{75}{137}$	$-\frac{12}{137}$	$\frac{60}{137}$
6	1	$-\frac{360}{147}$	$\frac{450}{147}$	$-\frac{400}{147}$	$\frac{225}{147}$	$-\frac{72}{147}$	$\frac{10}{147}$	$\frac{60}{147}$

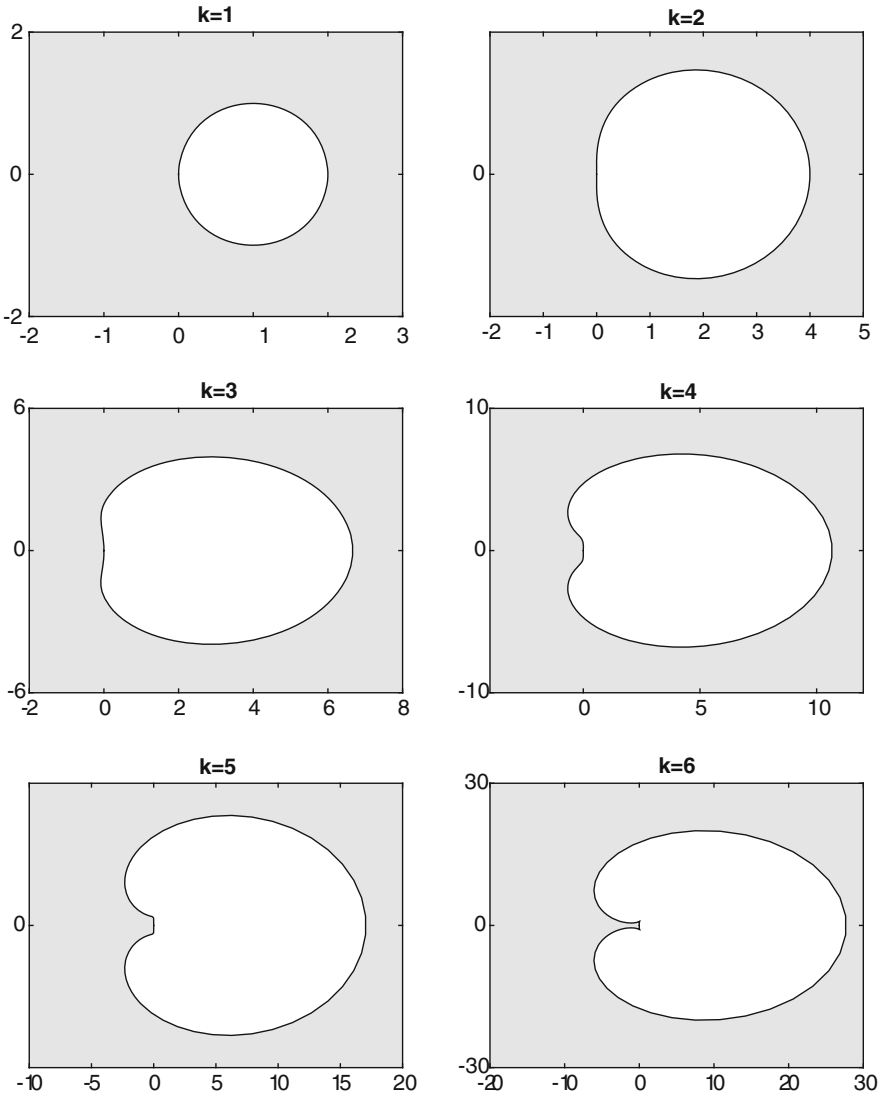


Fig. 7.1 Stability regions of the BDF methods (7.25) up to $k = 6$

Figure 7.2 depicts the subset $\mathcal{S}$ of the complex plane that is certainly contained in the stability region of a stiffly stable method. Certainly, all BDF methods are stiffly stable.

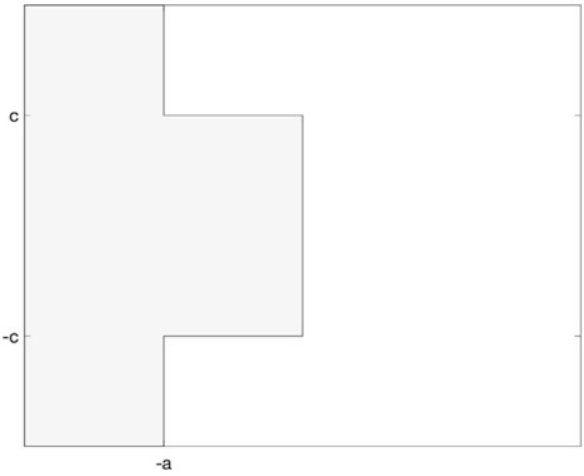


Fig. 7.2 Region of the complex plane contained in the stability region of a stiffly-stable method

7.6 Principles of Adaptive Integration

So far, we have considered discretizations of differential problems relying on fixed stepsize computational frameworks. Clearly, a-priori fixing the stepsize makes the numerical grid rigid: indeed, it creates a global discretization that reveals to be, in most of the cases, too coarse or too fine. It seems more reasonable to develop an adapted discretization that follows the behavior of the solution, modifying the stepsize in order to make it smaller only when necessary (e.g., when the solution rapidly changes its concavity, its monotony or when it shows fast oscillations).

We aim to briefly present here the principal steps behind the design of a variable stepsize numerical solver. More details can be found, for instance, in [19, 67, 124, 170–172, 195, 223, 228, 242, 243, 316–320, 323, 324].

The idea of adaptive numerical integration of ODEs by one-step methods can be outlined as follows:

- we focus on a single step from t_n to t_{n+1} and denote by h_n the amplitude of the interval $[t_n, t_{n+1}]$. We aim to compute y_{n+1} with a prescribed accuracy tol;
- once the numerical solution y_{n+1} is computed, we need to provide an estimation of the error in t_{n+1};
- if the error estimate does not exceed the prescribed accuracy tol, the computed value of y_{n+1} is accepted;
- if the error estimate exceeds tol, the computed value of y_{n+1} is rejected and recomputed with a smaller stepsize, until the error estimate becomes smaller than tol.

We understand from this outline that building blocks for the design of a variable stepsize computing framework certainly include a strategy to estimate the error,

a stepsize control strategy and, clearly, an effective way to handle implicitness in highly stable methods applied to stiff problems. We explain how to handle these issues for the classes of methods: one-step predictor-corrector numerical solvers and Runge-Kutta methods.

7.6.1 Predictor-Corrector Schemes

Predictor-corrector strategy is a well-known technique in the numerics for ODEs (see, for instance, [242]), useful to handle implicit methods avoiding a direct solution of the underlying nonlinear system of algebraic equations at each step.

Predictor-corrector schemes consist in coupling an explicit method (the so-called *predictor*) with an implicit method (denoted as *corrector*). Let us consider, for instance, such a scheme arising from coupling an explicit and an implicit LMM (3.1). Focusing on a single step to t_{n+k}:

- the explicit LMM computes a prediction of the solution in t_{n+k}, that we denote as y_{n+k}^{PRED};
- this prediction is included in the implicit LMM (3.5), as follows

$$y_{n+k}^{\text{CORR}} = h\beta_k f(t_{n+k}, y_{n+k}^{\text{PRED}}) + g_{n+k-1},$$

  in order to compute the corrected value y_{n+k}^{CORR}.

In other terms, having computed a predicted value to include in the corrector avoids the application of iterative methods for the solution of nonlinear systems, needed to handle the implicitness in direct way. Clearly, the effectiveness of this approach meets an expectable drawback: coupling an explicit and implicit methods certainly affects the stability of the overall scheme, since the explicit method has a bounded stability region; moreover, the order of convergence of the scheme is generally given by the minimum between the order of the predictor and that of the corrector [242].

We observe that the value of the corrector itself can also be iteratively corrected again, until the norm of the difference of two consecutive corrections is smaller than a certain tolerance. In this case, the scheme acts as follows:

- the explicit LMM computes a prediction of the solution in t_{n+k}, that we denote as y_{n+k}^{PRED};
- this prediction is included in the implicit LMM (3.5), as follows

$$y_{n+k}^{\text{CORR}} = h\beta_k f(t_{n+k}, y_{n+k}^{\text{PRED}}) + g_{n+k-1},$$

  in order to compute the first corrected value y_{n+k}^{CORR};

- the value of y_{n+k}^{CORR} is used to perform the following μ iterative corrections

$$y_{n+k}^{\text{CORR},[\nu]} = h\beta_k f\left(t_{n+k}, y_{n+k}^{\text{CORR},[\nu-1]}\right) + g_{n+k-1}, \quad \nu = 1, 2, \ldots, \mu,$$

with $y_{n+k}^{\text{CORR},[0]} = y_{n+k}^{\text{CORR}}$. This further correction is performed for μ iterations, such that $\|y_{n+k}^{\text{CORR},[\nu]} - y_{n+k}^{\text{CORR},[\nu-1]}\|_\infty$ is smaller than a certain tolerance.

Coupling a predictor and a corrector method does not only avoid the solution of nonlinear systems of algebraic equations at each step, but it also makes possible to provide an error estimate. Indeed, the comparison between the prediction and the correction has been used in the literature to construct the so-called *Milne error estimate* [242]. Supposing that two LMMs of order p are used as predictor and corrector formulae, we have

$$y(t_{n+k}) = y_{n+k}^{\text{PRED}} + C_{p+1}^\star h^{p+1} y^{(p+1)}(t_n) + O(h^{p+2}),$$
$$y(t_{n+k}) = y_{n+k}^{\text{CORR}} + C_{p+1} h^{p+1} y^{(p+1)}(t_n) + O(h^{p+2}),$$

where $C_{p+1}^\star$ is the error constant of the predictor and C_{p+1} that of the corrector. Side-by-side subtraction yields

$$y_{n+k}^{\text{PRED}} - y_{n+k}^{\text{CORR}} + \left(C_{p+1}^\star - C_{p+1}\right) h^{p+1} y^{(p+1)}(t_n) + O(h^{p+2}). \tag{7.26}$$

A direct application of the corrector method for the computation of y_{n+k} would give

$$y(t_{n+k}) - y_{n+k} = C_{p+1} h^{p+1} y^{(p+1)}(t_n) + O(h^{p+2}).$$

Replacing last expression in (7.26) leads to the Milne estimate of the error

$$\|y(t_{n+k}) - y_{n+k}\| \approx \left|\frac{C_{p+1}}{C_{p+1}^\star - C_{p+1}}\right| \|y_{n+k}^{\text{PRED}} - y_{n+k}^{\text{CORR}}\|. \tag{7.27}$$

Example 7.8 Let us compute Milne estimate (7.27) associated to the predictor-corrector scheme given by coupling explicit and implicit Euler methods (2.19)–(2.32). Since $C_2^\star = -C_2 = \frac{1}{2}$, we have

$$\|y(t_{n+1}) - y_{n+1}\| \approx \frac{1}{2}\|y_{n+1}^{\text{PRED}} - y_{n+1}^{\text{CORR}}\|. \tag{7.28}$$

(continued)

Example 7.8 (continued)
To check its accuracy, let us apply the explicit-implicit Euler predictor-corrector scheme to solve Prothero-Robinson problem (7.5) in [0, 10], with $g(t) = \sin(t)$, initial value $y_0 = 0$ and for $\lambda = -10$. The exact solution is $y(t) = \sin(t)$. Comparing Milne estimate (7.28) with the true error obtained as difference between the solution computed by the implicit Euler method and the exact solution leads to the following results: for $h = 0.01$, the infinity norm of the true error is $5.98 \cdot 10^{-4}$, while Milne estimate gives $5.55 \cdot 10^{-5}$; for $h = 0.005$, the infinity norm of the true error is $2.74 \cdot 10^{-4}$, while Milne estimate gives $1.31 \cdot 10^{-5}$. The values are pretty comparable, even if Milne estimate slightly underestimates the error.

7.6.2 Stepsize Control Strategies

As aforementioned, a variable stepsize computational environment requires, together with an error estimation strategy, the assessment of a technique of stepsize control. A classical stepsize control strategy (see, for instance, [198] and references therein) relies on the following observations.

Let us denote by $\|\text{est}(t_{n+1})\|$ the norm of the error estimate in t_{n+1}, obtained with stepsize h_n, and by tol the desired tolerance. Let us suppose that

$$\|\text{est}(t_{n+1})\| = \beta\, tol,$$

with $0 < \beta \leq 1$, if the error estimate is smaller or equal than the tolerance and $\beta > 1$ if the error estimate has not yet achieved the prescribed tolerance. Let us denote by h_{OPT} the largest stepsize we can use to achieve $\|\text{est}(t_{n+1})\| = tol$. Since $\|\text{est}(t_{n+1})\| \approx C_{p+1} h_n^{p+1} \|y^{(p+1)}(t_n)\|$, we have

$$\beta C_{p+1} h_{\text{OPT}}^{p+1} \|y^{(p+1)}(t_n)\| \approx C_{p+1} h_n^{p+1} \|y^{(p+1)}(t_n)\|$$

leading to

$$h_{\text{OPT}} \approx h_n \left(\frac{tol}{\|est(t_n)\|} \right)^{\frac{1}{p+1}}.$$

Taking into account this issue, a *classical stepsize controller* is given by

$$h_{n+1} = \text{fac} \cdot h_n \left(\frac{tol}{\|est(t_n)\|} \right)^{\frac{1}{p+1}}, \tag{7.29}$$

where fac is a security factor useful to avoid an uncontrolled stepsize growth (usually, it is chosen equal to 0.9).

This is a very basic stepsize controller, only depending on the current error estimate computed in the previous step, that often determines useless stepsize rejections, "*with disruptive and wasteful increases and decreases*"of the stepsize (see [67]). An improvement has been given by Gustafsson, Lundh and Söderlind [186, 323, 324], that introduced a different strategy, the so-called *PI stepsize control* , mainly based on control theory arguments. PI stepsize control involves the estimation of the local errors related to the two most recent subintervals of the discretization, i.e.,

$$h_{n+1} = h_n \cdot \min\left(2, \left(\frac{tol}{\|est(t_n)\|}\right)^{\sigma_1}\left(\frac{tol}{\|est(t_{n-1})\|}\right)^{\sigma_2}\right), \tag{7.30}$$

where σ_1 and σ_2 are parameters to be suitably chosen (see, for instance, [195, 323, 324].

The following program provides a variable stepsize implementation of the predictor-corrector scheme given by coupling explicit and implicit Euler methods (2.19)–(2.32), with Milne estimation of the error (7.28) and the classical stepsize control strategy (7.29).

Program 7.1 (Explicit-Implicit Euler Predictor-Corrector Scheme)

```
% Matlab script implementing the predictor-corrector
% scheme based on explicit and implicit Euler methods.
% The user is asked to provide the following inputs:
problem=input('Label of the problem: ');
tspan=input('Integration interval [t0,T]: ');
y0=input('Initial value: ');
h=input('Initial stepsize: ');
tol=input('Tolerance: ');

hAcc=[];          % vector of accepted stepsizes
hRej=[];          % vector of rejected stepsizes
tr=[];            % points of stepsize rejection
tt=[tspan(1)];    % points of stepsize acceptance
y=[y0];           % matrix storing the numerical solution
                  % at each step point (columnwise)
nval=0;           % number of function evaluations
mu=2;             % number of corrections at each step

% The entry condition in the while loop ensures that
% latest considered step point tt(end) falls in
% the integration interval.
while(abs(tspan(2)-tt(end))>5*eps)
      subint=[tt(end) tt(end)+h];
```

(continued)

Program 7.1 (continued)

```
        [yPC,est,nv]=PC(problem,subint,y(:,end),h,mu);
        nval=nval+nv;
        hopt=0.9*h*sqrt(tol/est);  % optimal stepsize

        if est<=tol % accepted step
           hAcc=[hAcc h];
           % h must not exceed the length of
           % the remaining part of the integration interval
           h=min(hopt,tspan(2)-tt(end));
           tt=[tt tt(end)+h];
           y=[y yPC];
           tt(end)
        else
           hRej=[hRej h];
           h=hopt;
           tr=[tr tt(end)];
        end
  end

  % Last step of legth tspan(2)-tt(end)
  if(tt(end)~=tspan(2))
     h=tspan(2)-tt(end);
     subint=[tt(end) tspan(2)];
     [yPC,est,nv]=PC(problem,subint,y(:,end),h,mu);
     nval=nval+nv;
     hAcc=[hAcc h];
     y=[y yPC];
  end

  fprintf('Number of accepted steps: %d \n', length(hAcc));
  fprintf('Number of rejected steps: %d \n', length(hRej));
  fprintf('Number of function evaluations: %d \n', nval);
  fprintf('Milne estimate in the endpoint: %2.4e \n', est);
  fprintf('Achieved correct digits: %2.4f \n', -log10(est));

  plot(tt,y)
  figure(2)
  semilogy(tt(2:end),hAcc,tr,hRej,'*r')

  % "True" error: difference between the PC solution minus
  % a reference solution computed by the built-in Matlab
  % function ode15s
  options=odeset('AbsTol',100*eps,'RelTol',100*eps);
  [tMat,yMat]=ode15s(@f,tspan,y0,options,problem);
  trueError=norm(y(:,end)'-yMat(end,:),'inf');
  fprintf('True error: %2.4e \n', trueError);
```

Above Matlab script makes use of the following function PC, computing a single step of the predictor-corrector

(continued)

Program 7.1 (continued)
scheme based on the explicit Euler method and the implicit Euler method (the latter applied μ times).

```
function [y1,est,nval]=PC(problem,tspan,y0,h,mu)
nval=0;
fPred=f(problem,tspan(1),y0);
yPred=y0+h*fPred;
nval=nval+1;
y1=yPred;
for j=1:mu
   fCorr=f(problem,tspan(2),y1);
   nval=nval+1;
   y1=y0+h*fCorr;
end
est=(norm(y1-yPred,'inf'))/2;   % Milne estimate
```

Example 7.9 Let us provide a numerical test based on the application of the explicit-implicit Euler predictor-corrector scheme to the Prothero-Robinson problem (7.5) in $[0,\ 10]$, with $g(t) = \sin(t)$, initial value $y_0 = 0$ and for various values of λ. Assuming $\lambda = -10$, $h_0 = 0.01$ and tol$= 10^{-6}$, Program 7.1 provides the following output:

```
Number of accepted steps: 5983
Number of rejected steps: 15
Number of function evaluations: 17994
Milne estimate in the endpoint: 8.1248e-07
Number of correct digits: 6.0902
True error: 7.9146e-04
```

From the analysis of the numerical results we appreciate that, in this case, the stepsize selection strategy looks efficient, since it rejects a very small number of stepsizes. However, Milne estimate looks rather optimistic, because the true error (computed assuming `ode15s` solution as a reference solution) is 1000 times larger. Figure 7.3 displays the pattern of the stepsize and highlights stepsize rejections: we can appreciate that the stepsize changes according to the behavior of the exact solution $y(t) = \sin(t)$ and rejection points correspond to the points of concavity change for the solution.

(continued)

Example 7.9 (continued)
Assuming $\lambda = -1000$, $h_0 = 0.01$ and tol$= 10^{-6}$, Program 7.1 provides the following output:

```
Number of accepted steps: 45825
Number of rejected steps: 13556
Number of function evaluations: 178143
Milne estimate in the endpoint: 5.5538e-07
Number of correct digits: 6.2554
True error: 8.2641e-05
```

In this case, the number of rejected steps is much larger and, as we can realize from Fig. 7.4, rejected steps are spread out overall the integration interval and, additionally, the pattern of the stepsize is very much oscillating (a zoomed portion of the graph is visible in Fig. 7.5). This is definitely not surprising: indeed, the problem is more stiff than in the case $\lambda = -10$ and, since the scheme involves and explicit method (explicit Euler), the stability region is bounded. So, the solver tries to find values of h such that $h\lambda$ remains in the stability region and, in order to succeed, it oscillates around its boundary.

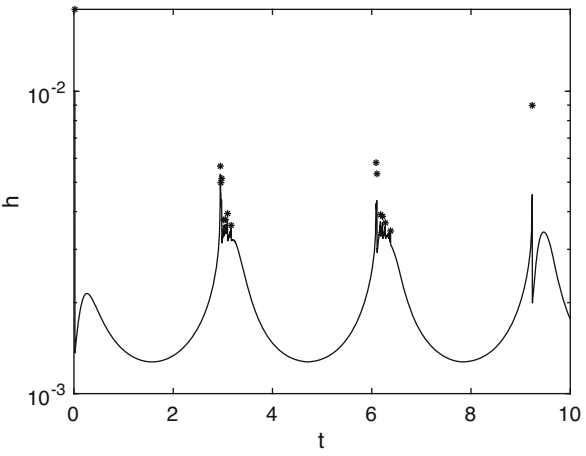


Fig. 7.3 Example 7.9: pattern of the stepsize associated to the application of the explicit-implicit Euler predictor-corrector scheme to the Prothero-Robinson problem (7.5) in $[0,\ 10]$, with $g(t) = \sin(t)$, initial value $y_0 = 0$ and $\lambda = -10$. The variable stepsize strategy used in Program 7.1 relies on Milne estimation of the error (7.28) and the classical stepsize control strategy (7.29). The initial stepsize is $h_0 = 0.01$ and the tolerance is tol$= 10^{-6}$. The stars highlight the rejected stepsizes

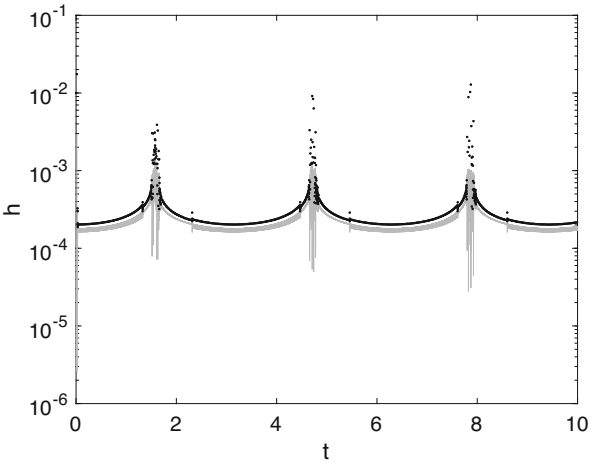


Fig. 7.4 Example 7.9: pattern of the stepsize associated to the application of the explicit-implicit Euler predictor-corrector scheme to the Prothero-Robinson problem (7.5) in $[0,\ 10]$, with $g(t) = \sin(t)$, initial value $y_0 = 0$ and $\lambda = -1000$. The variable stepsize strategy used in Program 7.1 relies on Milne estimation of the error (7.28) and the classical stepsize control strategy (7.29). The initial stepsize is $h_0 = 0.01$ and the tolerance is tol$= 10^{-6}$. The black stars highlight the rejected stepsizes

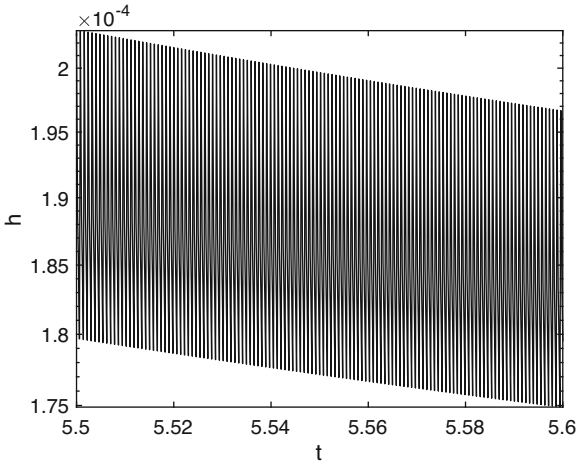


Fig. 7.5 Example 7.9: zoom of the pattern in Fig. 7.4 in a small interval

7.6.3 Error Estimation for Runge-Kutta Methods

Let us now briefly discuss some building blocks useful to design a variable stepsize solver based on RK methods (4.8), starting from the estimation of the error.

Many possibilities can be exploited, for instance the embedding strategy (here not discussed, but the reader can refer to [67, 195, 242], for instance). Here we discuss the so-called *Richardson extrapolation* strategy , based on applying the method twice, with stepsizes h and $2h$.

Let us focus on a fixed grid point t_{n+1} and denote by y_{n+1} and z_{n+1} RK solutions with stepsizes h and $2h$, respectively. Supposing that p is the order of the method, we have

$$y(t_{n+1}) = y_{n+1} + Ch^{p+1}y^{(p+1)}(t_n) + O(h^{p+2}),$$

$$y(t_{n+1}) = z_{n+1} + C(2h)^{p+1}y^{(p+1)}(t_n) + O(h^{p+2}),$$

where C is the error constant of the method. Side-by-side subtraction leads to

$$Ch^{p+1}y^{(p+1)}(t_n) \approx \frac{y_{n+1} - z_{n+1}}{2^{p+1} - 1}, \tag{7.31}$$

that is the estimate of the principal error term by the so-called Richardson extrapolation, that is known to be pretty accurate but expensive, since it requires two applications of the method.

Error estimates (such as Milne estimate or Richardson extrapolation) are asymptotically correct, i.e., when the stepsize tends to 0, by construction itself. As expectable, in order to approach stiff systems, this property of correctness may not be sufficient, since their solution also requires the usage of large stepsizes when the problem makes it possible. Shampine and Baca in [317] focused their attention on the assessment of the quality of the error estimate for large values of the stepsize, by using similar arguments as in the classical theory of absolute stability.

To this purpose, let us consider a restricted class of problems of the form $y' = Jy + g$, where J is a constant matrix and g a constant vector. Let us denote by $R(z)$ the rational function obtained by applying a given RK method to this problem, with z proportional to λ, and suppose that its error estimate can be represented as $R_e(hJ)(y_n - J^{-1}g)$. Moreover, we denote by $R_t(z) = \mathrm{e}^z - R(z)$. If

$$\frac{R_e(z)}{R_t(z)} \sim cz^m$$

for $\mathrm{Re}(z) < 0$ and $|z|$ tending to infinity, with positive integer m, so that the error is grossly overestimated for sufficiently large values of the stepsize.

In order to improve the error estimate in this case, the authors propose in [317] to multiply the estimate by the *filter matrix* $(I - hdJ)^{-1}$: for general ODEs (1.1), J is the Jacobian matrix of the vector field and d is a constant characteristic of the method. This choice is suitable to damp the large, stiff error components and, as observed in [317], the improved error estimator does not alter the behavior for small stepsizes and corrects it for large values of the stepsize.

7.6.4 Newton Iterations for Fully Implicit Runge-Kutta Methods

We complete this brief selection of topics useful to provide a variable stepsize implementation of RK methods by providing a representation of Newton iterations for fully implicit methods. As aforementioned, stiff problems generally have large Lipschitz constants making fixed-point iterations unsuitable to handle implicit methods. Let us now consider Newton iterations applied to the tensor representation of RK methods (4.11). We set

$$\Phi(Y) = Y - (e \otimes I)y_0 - h_0(A \otimes I)F(Y)$$

and aim to solve the system $\Phi(Y) = 0$, of dimension $sd \times sd$. We take as initial guess the vector

$$Y^{[0]} = \begin{bmatrix} y_0 \\ y_0 \\ \vdots \\ y_0 \end{bmatrix} \in \mathbb{R}^{sd},$$

and asses the following Newton iterative procedure

$$Y^{[\nu+1]} = Y^{[\nu]} - \left(\partial\Phi(Y^{[\nu]})\right)^{-1}\Phi(Y^{[\nu]}), \tag{7.32}$$

for $\nu \geq 1$, where

$$\partial\Phi(Y^{[\nu]}) = I_{sd} - h(A \otimes I_d)J(Y^{[\nu]}) \in \mathbb{R}^{sd\times sd}$$

and $J(Y^{[\nu]})$ is the Jacobian matrix of $F(Y^{[\nu]})$, i.e., the block diagonal matrix

$$J(Y^{[\nu]}) = \begin{bmatrix} \partial f(Y_1^{[\nu]}) & & \\ & \ddots & \\ & & \partial f(Y_s^{[\nu]}) \end{bmatrix},$$

where $\partial f(Y_j^{[\nu]})$ is the Jacobian matrix of f evaluated in $Y_j^{[\nu]}$, for $j = 1, 2, \ldots, s$. The matrix $\partial\Phi(Y^{[\nu]})$ is invertible for small enough values of h. The expression (7.32) is equivalent to the linear system

$$-\partial\Phi(Y^{[\nu]})\delta Y^{[\nu+1]} = \Phi(Y^{[\nu]}), \tag{7.33}$$

where $\delta Y^{[\nu+1]} = Y^{[\nu+1]} - Y^{[\nu+1]}$. We next solve the system (7.33) with respect to $\delta Y^{[0]}$, for example by Gaussian elimination, and derive

$$Y^{[0],i+1} = Y^{[0],i} + \delta Y^{[0]}.$$

We stop the iterative scheme at the M-th step, when $\|\delta Y^{[M]}\|_\infty$ is smaller than a prescribed tolerance and $\|\Phi(Y^{[M]})\|_\infty$ is also small enough. Then, we take $Y = Y^{[M]}$.

7.7 Exercises

1. Perform Prothero-Robinson analysis described in Sect. 7.2 to the two-stage Gaussian Runge-Kutta method (4.25). Comment the results.
2. Write a software in the programming language you prefer that provides a variable stepsize implementation of a BDF method, chosen among those provided in Sect. 7.5. The solver should incorporate the classical stepsize control (7.29) and Milne error estimate (7.27) to estimate the error.
3. Using the program developed in the previous exercise, provide an experimental confirmation of the stiff-stability of BDF methods.
4. Suppose that an explicit LMM (3.1) of order p and an implicit LMM of order q are coupled in a predictor-corrector scheme, where the corrector is iterated μ times per step. Prove that (see [242])
   - if $p \geq q$ (or if $p < q$ and $\mu > q - p$) the predictor-corrector scheme and the corrector method have the same order and the same principal error term;
   - if $p < q$ and $\mu = q - p$, the predictor-corrector scheme and the corrector method have the same order abut different principal error term;
   - if $p < q$ and $\mu \leq q - p - 1$, than the order of the predictor-corrector scheme is $p + \mu$. Provide an experimental confirmation of this accuracy property.
5. Provide a formal linear stability analysis of the predictor-corrector scheme given by coupling explicit and implicit Euler methods (2.19)–(2.32).
6. Find an empirical estimate of the parameters σ_1 and σ_2 in (7.30) to improve the performances of the variable stepsize implementation of the predictor-corrector scheme given by coupling explicit and implicit Euler methods (2.19)–(2.32). In particular, aim to improve the results in Fig. 7.4, trying to reduce the number of rejected steps. Comment the results.
7. Use Program 7.1 to solve the Brussellator problem [195]

$$\begin{cases} y_1'(t) = A + y_1^2 y_2 - (B+1) y_1(t), \\ \\ y_2'(t) = B y_1(t) - y_1(t)^2 y_2(t), \end{cases}$$

for $t \in [0, 20]$, with initial values $y_1(0) = 1.5$, $y_2(0) = 3$. Consider various tolerances and various values of the parameters A and B. Comment the results.

8. Use Program 7.1 to solve van der Pol problem [195]

$$\begin{cases} y_1'(t) = y_2(t), \\ \\ y_2'(t) = \big((1 - y_1(t)^2)y_2(t) - y_1(t)\big)/\varepsilon, \end{cases}$$

for $t \in [0, 2]$, with initial values $y_1(0) = 2$, $y_2(0) = -2/3$ and various values of the parameter ε (including 10^{-1}, 10^{-3} and 10^{-5}), observing that the problem is stiff for small values of ε. Comment the results.

9. Using the material covered in this chapter, write a software in the programming language you prefer that provides a variable stepsize implementation of Runge-Kutta methods, choosing an A-stable method. The solver should incorporate the classical stepsize control (7.29) and Richardson extrapolation (7.31) to estimate the error.

10. Compute the value of $\sigma_{\alpha,\beta}(S_0, I_0)$ defined by Eq. (7.4) for the countries listed in Table 1.2. Then, solve Eq. (7.3) by means of a chosen implicit method and check the number of time units needed to reach the maximum of infected people in the case of each country, observing that the highest values correspond to those countries exhibiting larger values of $\sigma_{\alpha,\beta}(S_0, I_0)$. Finally, solve the original nonlinear system (1.3) and compare the results previously obtained with its linearized version (7.3).

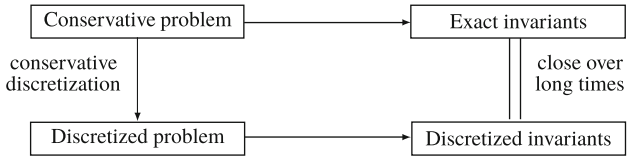
Chapter 8
Geometric Numerical Integration

It turned out that the preservation of geometric properties of the flow not only produces an improved qualitative behaviour, but also allows for a more accurate long-time integration than with general-purpose methods.

(Ernst Hairer, Christian Lubich, Gerhard Wanner, Preface of [192])

Modern Numerical Analysis is not only devoted to approximating the solutions of various problems through accurate and efficient numerical schemes, but also to retaining qualitative properties of the continuous problem over long times. Sometimes such conservation properties naturally characterize the numerical schemes, while in more complex situations preservation issues have to be conveyed into the numerical approximations. The numerical preservation of invariants is at the basis of the so-called *geometric numerical integration*. A classical reference to this topic is the monograph [192] by E. Hairer, C. Lubich and G. Wanner, which provides a comprehensive treatise on several aspects of geometric numerical integration.

The basic principle of geometric numerical integration can be briefly explained through the following diagram:



Indeed, suppose that a numerical method is applied to solve a conservative problem, i.e., a problem showing some invariants along the dynamics generated by its exact solution. A geometric numerical method provides a discretized problem that, along its solution, possesses invariants that are close to the exact ones over long time windows. Such a long-term preservation is not always automatically provided by any numerical method, hence it is relevant to analyze the conditions to impose

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_8

on a numerical scheme in order to make it a geometric numerical method. Before entering into the details of the topic, let us give an example.

Example 8.1 Let us consider the system of ODEs for the harmonic oscillator (1.20). As we have proved (see Example 1.7), the total energy (1.21) is a first integral of the system. We now aim to check if such a first integral remains invariant also along the numerical solutions computed by the following three methods:

- the explicit Euler method (2.19);
- the implicit Euler method (2.32);
- the two-stage Gaussian RK method (4.25).

Figures 8.1, 8.2 and 8.3 show the phase portrait of the approximate solutions to (1.20) with $\omega = 10$, computed over the time window $[0, 1000]$ by applying the aforementioned methods with constant stepsize 10^{-2}. As visible from these figures, both explicit and implicit Euler methods are not able to retain the symplecticity of the phase space, since they cannot reconstruct the periodic orbit characterizing the dynamics of (1.20). More specifically, the dynamics described by Fig. 8.1 is an outward spiral, due to the unstable behavior of the employed explicit method. On the contrary, the employ of an implicit method as in Fig. 8.2 yields an inward spiral dynamics. This is not the case of the two-stage Gaussian RK method (4.25) since, as visible from Fig. 8.3, it nicely maintains the symplecticity of the phase space.

A similar behavior can also be visible from the pattern of the deviation between the energy in the final integration point and that referred to the initial point. Indeed, Fig. 8.4 shows that the only method able to preserve the energy along time is the two-stage Gaussian RK method. The reason why this situation occurs will be clarified in the remainder of this chapter.

8.1 Historical Overview

The denomination *geometric numerical integration* strongly recalls the approach to geometry formulated by Felix Klein in his Erlangen program [238]. Klein describes geometry as the study of invariants under certain transformations. Similarly, geometric numerical methods were launched as structure-preserving schemes, able to retain peculiar features of a dynamical system along its discretizations. As addressed by Robert Mc Lachlan in his review [260] of the book by Hairer, Lubich and Wanner [192], the connection with the so-called *geometric integration theory* by Hassler Whitney [343] is even more subtle than that suggested by the name itself. Indeed, as

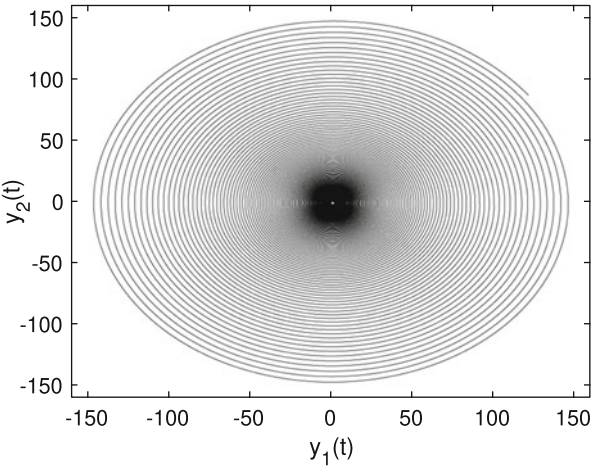


Fig. 8.1 Phase portrait of the approximate solution to the harmonic oscillator (1.20) with $\omega = 10$, initial values $y_1(0) = 0$ and $y_2(0) = 1$, computed by the Euler method (2.19) with stepsize 10^{-2}

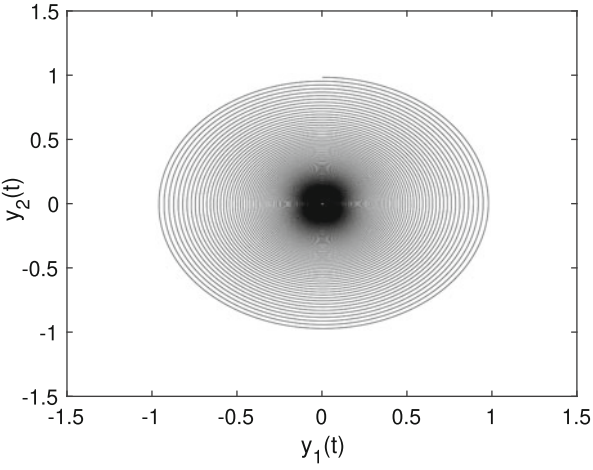


Fig. 8.2 Phase portrait of the approximate solution to the harmonic oscillator (1.20) with $\omega = 10$, initial values $y_1(0) = 0$ and $y_2(0) = 1$, computed by the implicit Euler method (2.32) with stepsize 10^{-2}

stated by Arnold [16] in his speech addressed to the participants of the International Congress of Mathematicians in Beijing, "*The design of stable discretizations of systems of PDEs often hinges on capturing subtle aspects of the structure of the system in the discretization. This new geometric viewpoint has provided a unifying understanding of a variety of innovative numerical methods developed over recent decades*". In his talk, Arnold shows that the function spaces introduced by Whitney in [343] (the so-called Whitney elements) represent what is required for a geometric discretization of many PDEs.

A famous method, well-known in the context of geometric numerical integration, is the so-called leapfrog method, also known as Störmer-Verlet method [192, 196].

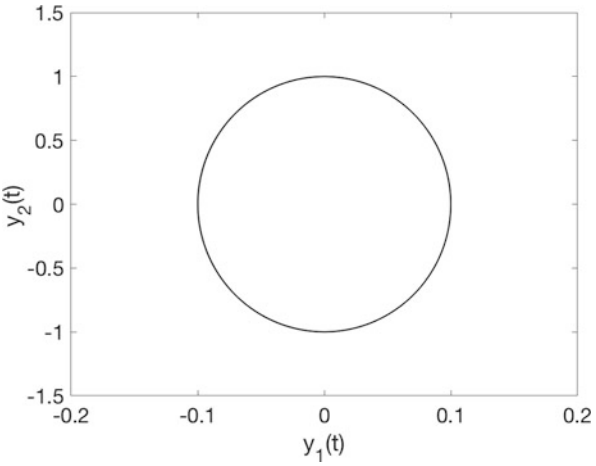


Fig. 8.3 Phase portrait of the approximate solution to the harmonic oscillator (1.20) with $\omega = 10$, initial values $y_1(0) = 0$ and $y_2(0) = 1$, computed by the two-stage Gaussian RK method (4.25) with stepsize 10^{-2}

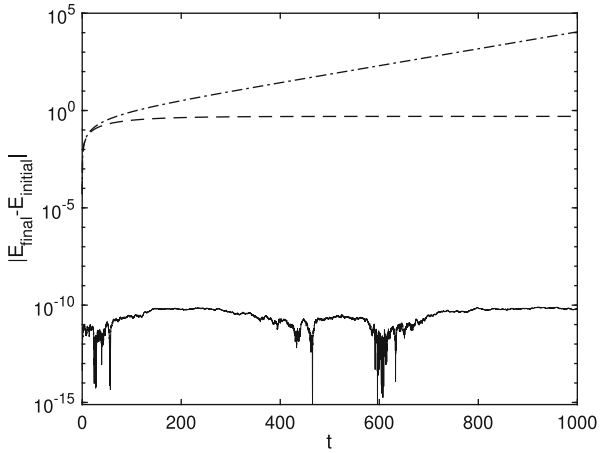


Fig. 8.4 Energy deviations in time along the approximate solutions to the harmonic oscillator (1.20) with $\omega = 10$, initial values $y_1(0) = 0$ and $y_2(0) = 1$, computed by the explicit Euler method (2.19, dashed-dotted line), the implicit Euler method (2.32, dashed line), the two-stage Gaussian RK method (4.25, solid line) with stepsize 10^{-2}. The deviation is computed as the absolute value of the difference between the energy in the final integration point $t = 1000$, minus that in the initial point $t = 0$

This method, for the discretization of the second order problem

$$\ddot{q} = f(q),$$

is given by

$$q_{n+1} - 2q_n + q_{n-1} = h^2 f(q_n).$$

This method is extensively used in many fields, such as celestial mechanics and molecular dynamics, and it is first due to Störmer that, in 1907, used a variant of this scheme for the computation of the motion of ionized particles in the Earth's magnetic field (aurora borealis). Above formulation is that developed by Verlet in 1967 [339] in his pioneering papers on the computer simulation of molecular dynamics models. Verlet was also particularly interested in the history of science, through which he was able to discover that his scheme was previously used by several authors (see [196] and references therein): for instance, by Delambre in 1792 for the computation of logarithms and astronomical tables (see [263]) and by Newton, who used it in his Principia (1687) to prove Kepler's second law (see [340]).

As highlighted in [196], a seminal contribution regarding geometric numerical integration was given by De Vogelaere in 1956 [144], "*a marvellous paper, short, clear, elegant, written in one week, submitted for publication and never published*". In particular, this paper provides examples of numerical methods (such as the symplectic Euler method) retaining the symplecticity of Hamiltonian problems. Still regarding Hamiltonian problems, successive contributions on their structure-preserving integrations are due to Ruth [305] in 1983 and Kang [232] in 1985.

A criterion for the numerical conservation of the symplecticity via Runge-Kutta methods (leading to the family of so-called *symplectic Runge-Kutta methods*) has independently been proved in 1988 by Lasagni [244], Sanz-Serna [307] and Suris [331], depending on a similar condition discovered by Cooper [98] for the numerical conservation of quadratic first integrals. To some extent, 1988 is the starting date for the spread out and the establishment of a theory of conservative numerical methods for Hamiltonian problems (on this topic, the interested reader can refer, for instance, to the monographs [26, 32, 192, 223, 233, 248, 249, 308], the survey papers [40, 41, 189, 261, 262, 264] and references therein).

Symplecticity is a prerogative of RK methods: in fact, Tang proved in 1993 [335] that linear multistep methods cannot be symplectic, as well as Hairer and Leone in 1997 [190, 250] and Butcher and Hewitt in 2009 [71] proved that genuine multivalue numerical methods cannot be symplectic. However, nearly-conserving linear multistep methods exhibiting excellent long-time behaviors have been developed by Hairer and Lubich [191, 192], Eirola and Sanz-Serna [158], while a theory of nearly-preserving multivalue methods has been explored in [67, 69, 70, 73, 122, 133, 134].

Other relevant classes of geometric numerical integrators fall in the field of the so-called *energy preserving* numerical integrators that are not considered here for the sake of brevity, but the interested reader can refer, for instance, to [31, 32, 34–36, 81–84, 92, 274–276, 294] and references therein.

This short historical overview of geometric numerical integration is clearly very far from being exhaustive and also the mentioned references are a small portion of the very wide scientific literature on the topic. However, it is in the author's opinion that even a brief glance at the historical frame is important to contextualize the results, better understand their genesis and the developments of new ideas.

8.2 Principles of Nonlinear Stability for Runge-Kutta Methods

We have introduced in Sect. 1.3 the relevant property of dissipativity of a differential problem, arising from a one-sided Lipschitz property of its vector field. In particular, we have proved that negative one-sided Lipschitz functions guarantee, according to Theorem 1.5, that contractive solutions with respect to a given norm are generated.

We now aim to understand under which conditions this feature is preserved along the solutions computed by a Runge-Kutta method, according to the following definition, given by Butcher in [61].

Definition 8.1 Let us consider a Runge-Kutta method applied to a differential problem (1.1) satisfying the contractivity condition

$$\langle f(t, y(t)) - f(t, \widetilde{y}(t)), y(t) - \widetilde{y}(t)\rangle \leq 0, \tag{8.1}$$

where $y(t)$ and $\widetilde{y}(t)$ are two solutions of (1.1), obtained with respect to the distinct initial values y_0 and $\widetilde{y}_0$, respectively. The method is *B-stable* if, for any stepsize h,

$$\|y_{n+1} - \widetilde{y}_{n+1}\| \leq \|y_0 - \widetilde{y}_0\|, \quad n \geq 0.$$

B-stable methods are certainly A-stable; this evidence can be proved by a simple check, obtained with respect to the Dahlquist test problem (6.1). The vice versa is not true. All Gaussian Runge-Kutta methods (see Sect. 4.4.1) are B-stable; the interested reader can find a detailed proof in [195].

Clearly, Definition 8.6 needs a practical way to check whether a Runge-Kutta method is B-stable or not. As usual, we present an algebraic condition on the coefficients of the method, ensuring its B-stability. Such a conditions has been independently proved by Burrage, Butcher [49] and Crouzeix [103].

Theorem 8.1 *For a given Runge-Kutta method (4.8), let us consider the matrix*

$$M = BA + A^{\intercal}B - bb^{\intercal}, \tag{8.2}$$

where $B = diag(b)$. If $b_i \geq 0$, $i = 1, 2, \ldots, s$ and M is non-negative definite, then the Runge-Kutta method is B-stable.

Proof According to Definition 8.6 of B-stability, let us consider a differential problem (1.1) generating contractive solutions and denote two of its solutions by $y(t)$ and $\widetilde{y}(t)$. Side-by-side subtraction between two applications of the Runge-Kutta method (4.8) for the approximation of $y(t)$ and $\widetilde{y}(t)$ yields

$$y_{n+1} - \widetilde{y}_{n+1} = y_n - \widetilde{y}_n + h \sum_{i=1}^{s} b_i \left(f(t_n + c_i h, Y_i) - f(t_n + c_i h, \widetilde{Y}_i) \right), \tag{8.3}$$

and

$$Y_i - \widetilde{Y}_i = y_n - \widetilde{y}_n + h \sum_{j=1}^{s} a_{ij} \left(f(t_n + c_j h, Y_j) - f(t_n + c_j h, \widetilde{Y}_j) \right). \tag{8.4}$$

Squaring side-by-side in (8.3) leads to

$$\begin{aligned} \|y_{n+1} - \widetilde{y}_{n+1}\|^2 = \|y_n - \widetilde{y}_n\|^2 & \\ +2h \sum_{i=1}^{s} b_i \langle f(t_n + c_i h, Y_i) - f(t_n + c_i h, \widetilde{Y}_i), y_n - \widetilde{y}_n \rangle & \\ +h^2 \sum_{i=1}^{s} \sum_{j=1}^{s} b_i b_j \langle f(t_n + c_i h, Y_i) - f(t_n + c_i h, \widetilde{Y}_i), & \\ f(t_n + c_j h, Y_j) - f(t_n + c_j h, \widetilde{Y}_j) \rangle. & \end{aligned}$$

Let us replace the value of $y_n - \widetilde{y}_n$ computed from (8.4) in the first scalar product appearing in the right-hand side of last equation, obtaining

$$\begin{aligned} \|y_{n+1} - \widetilde{y}_{n+1}\|^2 = \|y_n - \widetilde{y}_n\|^2 & \\ +2h \sum_{i=1}^{s} b_i \langle f(t_n + c_i h, Y_i) - f(t_n + c_i h, \widetilde{Y}_i), Y_i - \widetilde{Y}_i \rangle & \\ -h^2 \sum_{i=1}^{s} \sum_{j=1}^{s} m_{ij} \langle f(t_n + c_i h, Y_i) - f(t_n + c_i h, \widetilde{Y}_i), & \\ f(t_n + c_j h, Y_j) - f(t_n + c_j h, \widetilde{Y}_j) \rangle. & \end{aligned}$$

Taking into account the contractivity condition (8.1), the hypothesis $b_i \geq 0$, $i = 1, 2, \ldots, s$, and the characteristic property of non-negative matrices

$$\sum_{i=1}^{s} \sum_{j=1}^{s} m_{ij} \langle u_i, v_j \rangle \geq 0, \quad u_i, v_j \in \mathbb{R}^d, \quad i = 1, 2, \ldots, s,$$

the thesis holds true. □

Definition 8.2 A Runge-Kutta method (4.8) such that $b_i \geq 0$, $i = 1, 2, \ldots, s$, and whose matrix M defined by (8.2) is non-negative definite, is said to be *algebraically stable*.

According to Theorem 8.1 an algebraically stable RK method is B-stable. The vice versa is not true in general, unless the method is non-confluent, i.e., $c_i \neq c_j$, for any $i \neq j$. In this case, the following result holds true.

Theorem 8.2 *A non-confluent Runge-Kutta method is B-stable if and only if it is algebraically stable.*

The interested reader can find a complete proof of this result in [195]. An equivalence theorem for confluent methods has been proved by Hundsdorfer and Spijker in [220].

The concepts and the results contained in this section are a very brief introduction of the building blocks of the so-called *nonlinear stability* theory of numerical methods, i.e., the analysis of the properties of numerical methods applied to nonlinear problems and the ability of numerical discretizations to retain the qualitative properties of nonlinear test problems. Pioneering papers on nonlinear stability analysis for numerical methods approximating the solutions of ODEs have been provided by G. Dahlquist [110, 111], starting from the notion of G-stability (also see [65, 195]).

Let us now specialize our presentation to conservation issues for numerical methods approximating nonlinear problems with selected specific features.

8.3 Preservation of Linear and Quadratic Invariants

We have introduced the notion of first integral for a d-dimensional autonomous ODE (1.17) in Sect. 1.4. We now aim to analyze the conservative behavior of Runge-Kutta methods (4.8) if such a first integral is linear, i.e., it is of the form

$$I(y(t)) = v^{\intercal} y(t), \tag{8.5}$$

with $v \in \mathbb{R}^d$. The following result holds true.

Theorem 8.3 *Any Runge-Kutta method* (4.8) *preserves linear invariants* (8.5)*, i.e.,*

$$v^{\intercal} y_{n+1} = v^{\intercal} y_n, \quad n \geq 0.$$

Proof According to Definition 1.4, a first integral satisfies

$$\nabla I(y(t)) f(y(t)) = 0,$$

that means, for the linear case (8.5)

$$v^{\intercal} f(y(t)) = 0.$$

Let us compute $v^{\intercal} y_{n+1}$, where y_{n+1} is provided by a RK method (4.8), obtaining

$$v^{\intercal} y_{n+1} = v^{\intercal} y_n + h \sum_{i=1}^{s} b_i v^{\intercal} f(Y_i).$$

Since $v^{\intercal} f(Y_i) = 0$, $i = 1, 2, \ldots, s$, the thesis holds true. □

Let us now analyze the conservation of quadratic functions

$$Q(y(t)) = y(t)^{\intercal} C y(t), \tag{8.6}$$

where $C \in \mathbb{R}^{d \times d}$ is a symmetric matrix. Such a quadratic form is a first integral of (1.17), according to Definition 1.4, if

$$y(t)^{\intercal} C f(y(t)) = 0. \tag{8.7}$$

This condition is useful to prove the following result, proved by Cooper in [98].

Theorem 8.4 *If the coefficients of a Runge-Kutta method* (4.8) *fulfill the condition*

$$b_i a_{ij} + b_j a_{ji} = b_i b_j, \quad i, j = 1, 2, \ldots, s, \tag{8.8}$$

then it preserves quadratic invariants (8.6)*, i.e.,*

$$y_{n+1}^{\intercal} C y_{n+1} = y_n^{\intercal} C y_n, \quad n \geq 0.$$

Proof Let us compute the quadratic form $y_{n+1}^{\intercal} C y_{n+1}$, obtaining

$$y_{n+1}^{\intercal} C y_{n+1} = y_n^{\intercal} C y_n + h \sum_{i=1}^{s} b_i f(Y_i)^{\intercal} C y_n + h \sum_{i=1}^{s} b_i y_n^{\intercal} C f(Y_i)$$
$$+ h^2 \sum_{i,j=1}^{s} b_i b_j f(Y_i)^{\intercal} C f(Y_j).$$

Let us analyze the $O(h)$ terms in the right-hand side of last equation, by recasting y_n using the formula of the internal stages in (4.8), i.e.,

$$y_n = Y_i - h \sum_{j=1}^{s} a_{ij} f(Y_j).$$

We correspondingly obtain

$$h \sum_{i=1}^{s} b_i f(Y_i)^{\intercal} C y_n = h \sum_{i=1}^{s} b_i f(Y_i)^{\intercal} C Y_i - h^2 \sum_{i,j=1}^{s} b_i a_{ij} f(Y_i)^{\intercal} C f(Y_j),$$
$$h \sum_{i=1}^{s} b_i y_n^{\intercal} C f(Y_i) = h \sum_{i=1}^{s} b_i Y_i^{\intercal} C f(Y_i) - h^2 \sum_{i,j=1}^{s} b_j a_{ji} f(Y_i)^{\intercal} C f(Y_j),$$

i.e., by means of (8.7),

$$h \sum_{i=1}^{s} b_i f(Y_i)^{\intercal} C y_n = -h^2 \sum_{i,j=1}^{s} b_i a_{ij} f(Y_i)^{\intercal} C f(Y_j),$$
$$h \sum_{i=1}^{s} b_i y_n^{\intercal} C f(Y_i) = -h^2 \sum_{i,j=1}^{s} b_j a_{ji} f(Y_i)^{\intercal} C f(Y_j).$$

We finally get

$$y_{n+1}^{\intercal} C y_{n+1} = y_n^{\intercal} C y_n - h^2 \sum_{i,j=1}^{s} (b_i a_{ij} + b_j a_{ji} - b_i b_j) f(Y_i)^{\intercal} C f(Y_j),$$

leading to the thesis. □

It is worth observing that Eq. (8.8) provides an algebraic condition on the coefficients of RK methods that can more compactly be written as $M = 0$, where the matrix M is defined by (8.2). In other terms, the matrix M plays a role both in retaining the contractive character of solutions to dissipative problems and in conserving quadratic first integrals. However, the story does not end here, as we recognize in next section: indeed, RK methods satisfying (8.8) are particularly relevant in the numerical approximation of Hamiltonian problems.

We have realized that any Runge-Kutta method is able to exactly preserve linear invariants, while quadratic invariants are preserved only by a family of Runge-Kutta methods. A natural question to ask is what happens to polynomial invariants of degree greater than or equal to 3. This (negative) result gives the answer related to RK methods, whose complete proof can be found in [192]. Clearly, as aforementioned, since Runge-Kutta methods are not able to cover themselves all possible conservation issues, other relevant classes of geometric numerical integrators have been introduced, most of them falling in the general field of *energy-preserving* numerical methods (the reader can refer, for instance, to [31, 32, 34–36, 81–84, 92, 274–276, 294] and references therein).

8.4 Symplectic Methods

We have introduced a relevant class of conservative problems in Sect. 1.4, i.e., Hamiltonian problems (1.28). A characteristic property of these problems, as proved in Theorem 1.6 is the symplecticity of the corresponding flow map. In the spirit of geometric numerical integration we are interested in understanding under which conditions a numerical method is able to retain the same property along discretized dynamics. Let us particularly focus on one-step methods; we represent them as a map φ_h that associates y_{n+1} to y_n and give the following definition.

Definition 8.3 A one-step method is *symplectic* if the one-step map φ_h is a symplectic transformation when applied to a smooth Hamiltonian problem (1.28), i.e., if

$$\varphi_h'(y_n)^\intercal J \varphi_h'(y_n) = J.$$

We now provide important examples of symplectic methods, starting from the famous *symplectic Euler method*, introduced by de Vogelaere in [144].

Theorem 8.5 (de Vogelaere) *The* symplectic Euler method

$$\begin{aligned} p_{n+1} &= p_n - h\mathcal{H}_q(p_{n+1}, q_n), \\ q_{n+1} &= q_n + h\mathcal{H}_p(p_{n+1}, q_n), \end{aligned} \tag{8.9}$$

for the numerical solution of Hamiltonian problems (1.22) *is a symplectic method of order* 1.

Proof We first differentiate (8.9) side-by-side with respect to (p_n, q_n), obtaining

$$\begin{aligned} \frac{\partial p_{n+1}}{\partial p_n} &= \frac{\partial p_n}{\partial p_n} - h\mathcal{H}_{qp}\frac{\partial p_{n+1}}{\partial p_n}, \\ \frac{\partial p_{n+1}}{\partial q_n} &= -h\mathcal{H}_{qp}\frac{\partial p_{n+1}}{\partial q_n} - h\mathcal{H}_{qq}\frac{\partial q_n}{\partial q_n}, \\ \frac{\partial q_{n+1}}{\partial p_n} &= h\mathcal{H}_{pp}\frac{\partial p_{n+1}}{\partial p_n} \\ \frac{\partial q_{n+1}}{\partial q_n} &= \frac{\partial q_n}{\partial q_n} + h\mathcal{H}_{pp}\frac{\partial p_{n+1}}{\partial q_n} + h\mathcal{H}_{pq}\frac{\partial q_n}{\partial q_n} \end{aligned}$$

being $I \in \mathbb{R}^{d\times d}$ the identity matrix and avoiding to explicitly write the dependence of the Hamiltonian function on (p_{n+1}, q_n) for the sake of brevity. As a consequence,

$$\begin{aligned} \left(I + h\mathcal{H}_{qp}\right)\frac{\partial p_{n+1}}{\partial p_n} &= I, \\ \left(I + h\mathcal{H}_{qp}\right)\frac{\partial p_{n+1}}{\partial q_n} &= -h\mathcal{H}_{qq}, \\ -h\mathcal{H}_{pp}\frac{\partial p_{n+1}}{\partial p_n} + \frac{\partial q_{n+1}}{\partial p_n} &= 0, \\ -h\mathcal{H}_{pp}\frac{\partial p_{n+1}}{\partial q_n} + \frac{\partial q_{n+1}}{\partial q_n} &= I + h\mathcal{H}_{pq}. \end{aligned}$$

Recasting above relations in a compact matrix form yields

$$\begin{bmatrix} I + h\mathcal{H}_{qp} & 0 \\ -h\mathcal{H}_{pp} & I \end{bmatrix} \begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = \begin{bmatrix} I & -h\mathcal{H}_{qq} \\ 0 & I + h\mathcal{H}_{pq} \end{bmatrix},$$

from which we compute

$$\begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = \begin{bmatrix} I + h\mathcal{H}_{qp} & 0 \\ -h\mathcal{H}_{pp} & I \end{bmatrix}^{-1} \begin{bmatrix} I & -h\mathcal{H}_{qq} \\ 0 & I + h\mathcal{H}_{pq} \end{bmatrix}$$

$$= \begin{bmatrix} D & -hD\mathcal{H}_{qq} \\ h\mathcal{H}_{pp}D & -h^2\mathcal{H}_{pp}D\mathcal{H}_{qq} + D^{-1} \end{bmatrix},$$

where $D = (I + h\mathcal{H}_{qp})^{-1}$. The reader can easily check that the symplecticity condition

$$\begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix}^{\intercal} J \begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = J$$

holds true. □

We observe that the symplectic Euler method (8.9) is implicit with respect to p. An alternative version implicit in q also exists, given by

$$\begin{aligned} p_{n+1} &= p_n - h\mathcal{H}_q(p_n, q_{n+1}), \\ q_{n+1} &= q_n + h\mathcal{H}_p(p_n, q_{n+1}) \end{aligned} \tag{8.10}$$

and the reader can check its symplecticity, applying similar arguments as those used in the proof of Theorem 8.5, see Exercise 1 at the end of this chapter.

Let us now provide a Matlab implementation of the symplectic Euler method (8.9) applied to (1.22), given in Program 8.1. The code requires defining the right-hand side of (1.22) through the functions `fp.m` and `fq.m`. Moreover, the built-in function `fsolve` is used to handle the implicitness of (8.9).

Program 8.1 (Symplectic Euler Method)

```
% Function implementing the symplectic Euler method (8.9)
% for the numerical solution of a Hamiltonian problem
% on a uniform grid.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
```

(continued)

Program 8.1 (continued)

```
% - p0: initial momentum;
% - q0: initial position;
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - p: d×N matrix whose i-th column p(:,i) stores the
%      approximate momentum in the i-th grid point;
% - q: d×N matrix whose i-th column p(:,i) stores the
%      approximate position in the i-th grid point.

function [t,p,q]=symplecticEuler(problem,tspan,p0,q0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(p0);
p=zeros(d,N);
q=zeros(d,N);
options=optimset('Display','off','TolFun',eps,'TolX',eps);
p(:,1)=fsolve(@(x) x-p0-h*fp(problem,x,q0),p0,options);
q(:,1)=q0+h*fq(problem,p(:,1),q0);
for i=2:N
    p(:,i)=fsolve(@(x) x-p(:,i-1)+...
           h*fp(problem,x,q(:,i-1)),p(:,i-1),options);
    q(:,i)=q(:,i-1)+h*fq(problem,p(:,i),q(i-1));
end
```

Example 8.2 Let us solve the system of ODEs for the mathematical pendulum (1.23) by the symplectic Euler method (8.9), in order to check if the symplecticity of the continuous flow is also retained along the numerical dynamics. The numerical evidence is provided by using Program 8.1 and displayed in Fig. 8.5, showing that the symplecticity of the phase space is nicely preserved by (8.9) that provides the periodic orbit characterizing the dynamics of (1.23). This property is not visible if a non-symplectic method is used: for instance, computing the numerical dynamics by means of the explicit Euler method (2.19) provides the phase portrait depicted in Fig. 8.6, where the symplecticity of the original problem is totally lost.

Let us now analyze the property of symplecticity for Runge-Kutta methods, applied to Hamiltonian problems (1.22). This topic has been object of seminal papers, all dated 1988, independently authored by Lasagni [244], Sanz-Serna [307], Suris [331]. The proof of symplecticity for Runge-Kutta methods relies on the following lemma [27, 192].

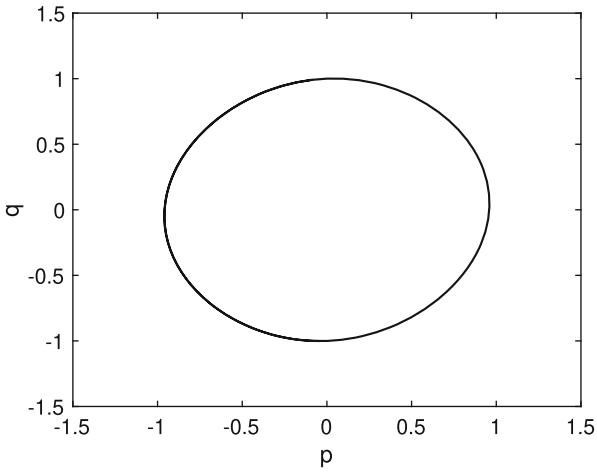


Fig. 8.5 Phase portrait associated to the approximate solution to the mathematical pendulum (1.23) with initial values $p(0) = 0$ and $q(0) = 1$, computed by the symplectic Euler method (8.9) with stepsize 10^{-1}

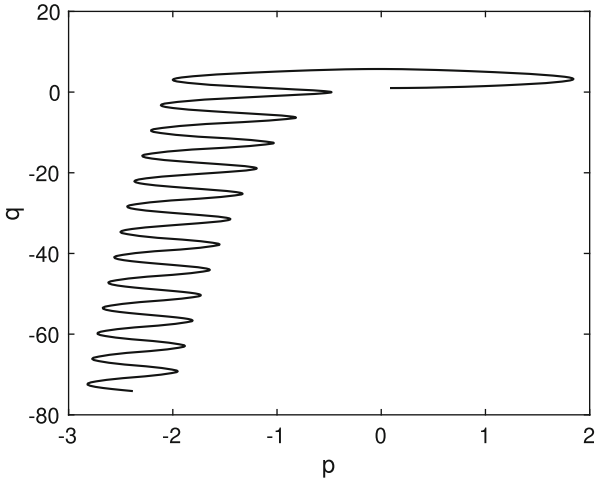
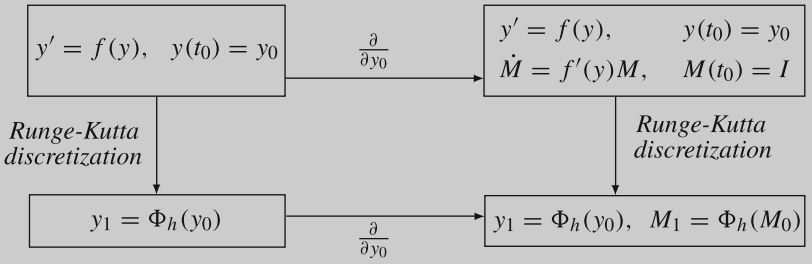


Fig. 8.6 Phase portrait associated to the approximate solution to the mathematical pendulum (1.23) with initial values $p(0) = 0$ and $q(0) = 1$, computed by the explicit Euler method (2.19) with stepsize 10^{-1}

Lemma 8.1 *Consider an autonomous problem* (1.17) *and its variational equation* (1.29). *Correspondingly, let us denote by* $y_{n+1} = \Phi_h(y_n)$ *the map associating a single step of a given Runge-Kutta method from the point* t_n *to* t_{n+1} *of the grid. Then, the following diagram commutes:*

(continued)

Lemma 8.1 (continued)



where the horizontal arrows denote differentiation with respect to y_0 and the vertical arrows the application of Φ_h. In other terms, the numerical result $\{y_1, M_1\}$ obtained by applying a single step of the method to the problem augmented by its variational equation is equal to the numerical solution of $\dot{y} = f(y)$ augmented by its derivative $M_1 = \partial y_1/\partial y_0$.

Proof We first compute a single step of a RK method (4.8) applied to (1.17) and side-by-side differentiate with respect to y_0, obtaining

$$\begin{aligned} \frac{\partial y_1}{\partial y_0} &= I + h \sum_{i=1}^{s} b_i f'(Y_i) \frac{\partial Y_i}{\partial y_0}, \\ \frac{\partial Y_i}{\partial y_0} &= I + h \sum_{j=1}^{s} a_{ij} f'(Y_j) \frac{\partial Y_j}{\partial y_0}, \quad i = 1, 2, \ldots, s. \end{aligned} \tag{8.11}$$

We observe that the last equation is a linear system in the unknowns $\frac{\partial Y_i}{\partial y_0}$, $i = 1, 2, \ldots, s$.

We now aim to prove that side-by-side differentiating (1.17) and then applying (4.8) lead to the same result. So, we apply (4.8) directly to the variational equation (1.29), getting

$$\begin{aligned} \frac{\partial y_1}{\partial y_0} &= I + h \sum_{i=1}^{s} b_i f'(Y_i) \widetilde{M}_i, \\ \widetilde{M}_i &= I + h \sum_{j=1}^{s} a_{ij} f'(Y_j) \widetilde{M}_j, \quad i = 1, 2, \ldots, s. \end{aligned} \tag{8.12}$$

We observe that last equation is also a linear system in the unknowns $\widetilde{M}_i$, $i = 1, 2, \ldots, s$. Moreover, the two linear systems displayed as second equations of (8.11) and (8.12) act exactly in the same way. For sufficiently small values of h, both systems have unique solution and, since they are the same system, we have $\widetilde{M}_i = \partial Y_i / \partial y_0$ and, consequently, $M_1 = \partial y_1 / \partial y_0$. So the diagram in the statement of the lemma commutes. □

Theorem 8.6 *Any RK method* (4.8) *preserving quadratic first integrals* (8.6) *is a symplectic method.*

Proof Let us consider the augmented system

$$\begin{aligned} \dot{y} &= J^{-1}\nabla\mathcal{H}(y), \\ \dot{M} &= J^{-1}\nabla^2\mathcal{H}(y)M, \end{aligned} \tag{8.13}$$

containing the Hamiltonian problem (1.28) and its variational equation. Let us prove that $M^\intercal J M$ is a first integral for (8.13). Indeed,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}(M^\intercal J M) &= \dot{M}^\intercal J M + M^\intercal J \dot{M} \\ &= \left(J^{-1}\nabla^2\mathcal{H}(y)M\right)^\intercal J M + M^\intercal J J^{-1}\nabla^2\mathcal{H}(y)M \\ &= M^\intercal \left(\nabla^2\mathcal{H}(y)\right)^\intercal (J^{-1})^\intercal J\psi + M^\intercal\nabla^2\mathcal{H}(y)M \\ &= -M^\intercal\nabla^2\mathcal{H}(y)M + M^\intercal\nabla^2\mathcal{H}(y)M = 0. \end{aligned}$$

In other terms, $M^\intercal J M$ is a quadratic first integral of (8.13) and is preserved by any RK method fulfilling the condition (8.8) of conservation of quadratic invariants described in Theorem 8.4. The conserved value of $M^\intercal J M$ is then equal to its initial value, i.e., $M^\intercal J M = J$, that is the symplecticity condition. So, all RK conserving quadratic invariants are symplectic. □

It is worth highlighting that condition (8.8) is then also a symplecticity condition. For this reason, the literature directly denotes RK methods satisfying (8.8) as *symplectic RK methods*. A consequence of this result is that all Gaussian RK methods (see Sect. 4.4.1) are symplectic methods; Program 8.2 implements one of them, namely that depending on two internal stages (4.25), to solve a given Hamiltonian problem.

Program 8.2 (Symplectic RK Method (2-Stage Gaussian Method))

```
% Function implementing the 2-stage Gaussian method (4.25)
% for the numerical solution of a Hamiltonian problem
% on a uniform grid.

% Inputs
% - problem: label of the problem to be solved;
% - tspan: vector of two components, storing the extrema
%          of the integration interval;
% - y0: vector of initial momenta (stored in y0(1:d)) and
%       initial positions (stored in y0(d+1:2d))
% - h: constant stepsize.

% Outputs
% - t: set of N equidistant grid points (tspan(1) excluded);
% - y: 2d×N matrix whose i-th column stores approximate
%      momenta (in y0(1:d)) and coordinates (in y0(d+1:2d)),
%      referring to the i-th grid point;
% - hamDev: N-dimensional vector storing the deviation
%          of the Hamiltonian function in each grid point
%          from the initial Hamiltonian;

function [t,y,hamDev]=GaussRK2s(problem,tspan,y0,h)
N=(tspan(2)-tspan(1))/h;
t=linspace(h,tspan(2),N);
d=length(y0)/2;
Id=eye(2*d);
y=zeros(2*d,N);
hamDev=zeros(N,1);
c=[(3-sqrt(3))/6; (3+sqrt(3))/6]; e=ones(length(c),1);
A=[1/4 1/4-sqrt(3)/6; 1/4+sqrt(3)/6 1/4];
b=[1; 1]/2;
options=optimset('Display','off','TolFun',eps,'TolX',eps);
Y=fsolve(@(Z) Z-kron(e,Id)*y0-h*kron(A,Id)*...
  [f(problem,[],Z(1:2*d)); f(problem,[],Z(2*d+1:4*d))],...
  [y0; y0],options);
y(:,1)=y0+h*kron(b',Id)*...
       [f(problem,[],Y(1:2*d)); f(problem,[],Y(2*d+1:4*d))];
ham0=hamiltonian(problem,y0);
hamDev(1)=abs(hamiltonian(problem,y(:,1))-ham0);
for i=2:N
    Y=fsolve(@(Z) Z-kron(e,Id)*y(:,i-1)-h*kron(A,Id)*...
      [f(problem,[],Z(1:2*d)); f(problem,[],Z(2*d+1:4*d))],...
      [y(:,i-1); y(:,i-1)],options);
    y(:,i)=y(:,i-1)+h*kron(b',Id)*[f(problem,[],Y(1:2*d));
          f(problem,[],Y(2*d+1:4*d))];
    hamDev(i)=abs(hamiltonian(problem,y(:,i))-ham0);
end
```

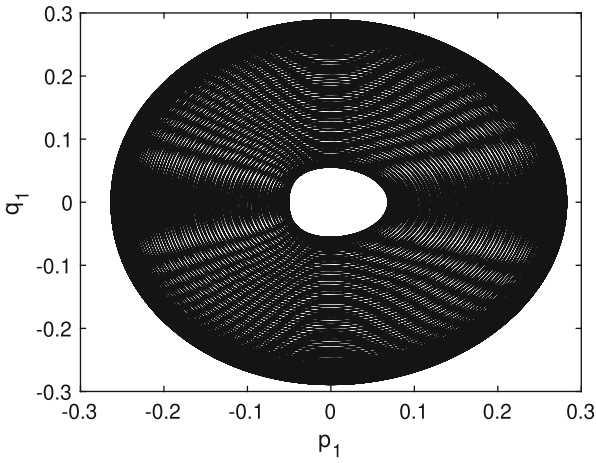


Fig. 8.7 Phase portrait of (1.27) in the (p_1, q_1)-plane, with initial values $p_1(0) = 0.2$, $p_2(0) = 0$, $q_1(0) = -0.2$, $q_2(0) = 0$. The displayed dynamics originates from the application of the symplectic two-stage Gaussian RK method (4.25) with stepsize $h = 0.1$

A numerical evidence of the symplecticity of Gaussian RK method is certainly given by Example 8.1. An additional one is reported in the following example, whose results have been obtained via Program 8.2.

Example 8.3 Let us consider Hénon-Heiles problem (1.26), already analyzed in Example 1.9 in order to provide a numerical evidence of the symplecticity of the two-stage Gaussian RK method (4.25). Figures 8.7, 8.8, 8.9, and 8.10 display the phase portrait in several planes and provide a confirmation of the symplecticity of the numerical scheme, able to recover the symplecticity of the original problem along the numerical dynamics. We observe that the chosen time window is [0, 4000] and the employed stepsize is $h = 0.1$.

8.5 Symmetric Methods

A relevant property of mechanical systems is their time reversibility; in terms of flow map, this property is equivalent to say that $\Phi_t \circ \Phi_{-t}$ is the identity map. In other terms, for a reversible system with initial value y_0, the dynamics starting from $y(t)$ with reverse time goes back to y_0. In this section, we aim to understand under which conditions this property is recovered by a one-step method. Then, the following definitions are given.

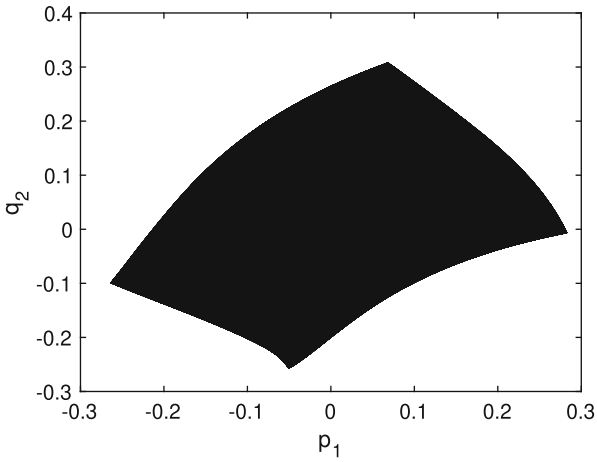


Fig. 8.8 Phase portrait of (1.27) in the (p_1, q_2)-plane, with initial values $p_1(0) = 0.2$, $p_2(0) = 0$, $q_1(0) = -0.2$, $q_2(0) = 0$. The displayed dynamics originates from the application of the symplectic two-stage Gaussian RK method (4.25) with stepsize $h = 0.1$

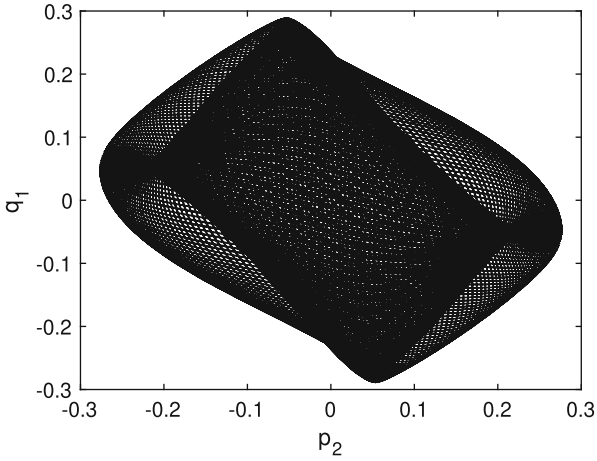


Fig. 8.9 Phase portrait of (1.27) in the (p_2, q_1)-plane, with initial values $p_1(0) = 0.2$, $p_2(0) = 0$, $q_1(0) = -0.2$, $q_2(0) = 0$. The displayed dynamics originates from the application of the symplectic two-stage Gaussian RK method (4.25) with stepsize $h = 0.1$

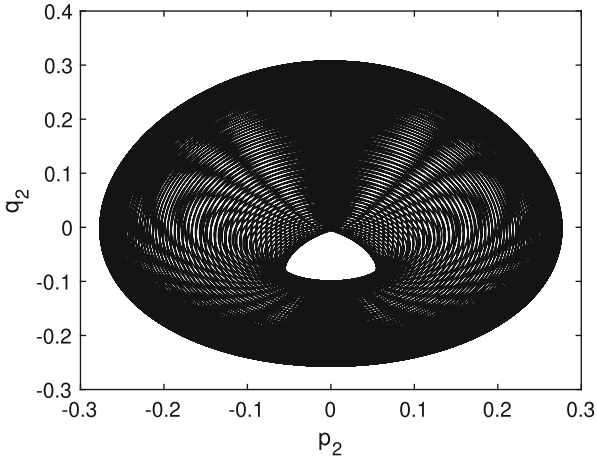


Fig. 8.10 Phase portrait of (1.27) in the (p_2, q_2)-plane, with initial values $p_1(0) = 0.2$, $p_2(0) = 0$, $q_1(0) = -0.2$, $q_2(0) = 0$. The displayed dynamics originates from the application of the symplectic two-stage Gaussian RK method (4.25) with stepsize $h = 0.1$

Definition 8.4 Given a one-step method φ_h, its *adjoint method* is the one-step map

$$\varphi_h^\star = \varphi_{-h}^{-1}.$$

Definition 8.5 A one-step method φ_h is *symmetric* if it is equal to its adjoint.

Example 8.4 Let us compute the adjoint of the explicit Euler method (2.19), i.e.,

$$y_n = y_{n+1} - hf(y_{n+1}).$$

Rearranging the terms in the last equation leads to the implicit Euler method (2.32). Hence, the explicit Euler method is not self-adjoint, so it is not symmetric.

The implicit midpoint method (4.24) is symmetric since its adjoint method is given by

$$y_n = y_{n+1} - hf\left(\frac{1}{2}(y_{n+1} + y_n)\right),$$

i.e., it is the implicit midpoint method as well.

The following theorem provides a relevant accuracy property of symmetric methods, useful for their construction and analysis. Indeed, we now prove that the order of convergence of a symmetric method is always even, then their construction requires to fulfill a restricted number of order conditions.

Theorem 8.7 *The order of a symmetric one-step method is even.*

Proof Let us denote by p the order of convergence of the method. Then (also see Theorem 3.2, Section II.3 in [192]), a single step of length h satisfies

$$\varphi_h(y_0) = \Phi_h(y_0) + Ch^{p+1} + O(h^{p+2}),$$

where C is the error constant of the method. Performing a step in reverse time leading to y_0 yields

$$y_0 = \varphi_{-h}(\Phi_h(y_0)) + (-1)^p Ch^{p+1} + O(h^{p+2}).$$

Inverting the operator

$$\varphi_h^\star(y_0) = \Phi_h(y_0) + (-1)^p Ch^{p+1} + O(h^{p+2}).$$

Therefore, the adjoint of a method of order p has order p as well. Moreover, since the method is symmetric, then $C = (-1)^p C$ and, as a consequence, the error constant C is different from 0 only for even values of p. □

We now aim to give a characterization of symmetric Runge-Kutta methods, provided in terms of algebraic conditions on their coefficients, as usual.

Theorem 8.8 *If the coefficients of a given Runge-Kutta method* (4.8) *satisfy the conditions*

$$a_{s+1-i,s+1-j} + a_{ij} = b_j, \quad i, j = 1, 2, \ldots, s, \tag{8.14}$$

then, the method is symmetric.

Proof The first step of the proof consists in computing the coefficients of the adjoint of a Runge-Kutta method (4.8). Referring to a single step with stepsize $-h$, leading to y_n if we start from y_{n+1}, the internal stages $Y_i^\star$ of the adjoint method are given by

$$\begin{aligned} Y_i^\star &= y_{n+1} - h\sum_{j=1}^{s} a_{ij} f(Y_j) = y_n + h\sum_{j=1}^{s} b_j f(Y_j) - h\sum_{j=1}^{s} a_{ij} f(Y_j) \\ &= y_n + h\sum_{j=1}^{s} (b_j - a_{ij}) f(Y_j). \end{aligned}$$

Observing that the internal stages of the adjoint method appear in reverse order with respect to those of the original method, i.e.,

$$Y_i^\star = Y_{s+1-i}, \quad i = 1, 2, \ldots, s,$$

the coefficients of the adjoint are then given by

$$a_{ij}^\star = b_{s+1-j} - a_{s+1-i,s+1-j}, \quad i, j = 1, 2, \ldots, s.$$

Proceeding similarly with the advancing law, we obtain

$$b_i^\star = b_{s+1-i}, \quad i = 1, 2, \ldots, s.$$

The second step of the proof is a trivial check of the conditions guaranteeing that the method is equal to its adjoint, i.e., $a_{ij}^\star = a_{ij}$ and $b_i^\star = b_i$, leading to the thesis. □

Example 8.5 Let us specialize the symmetry conditions (8.14) to specific values of s, in order to check the symmetry of some methods presented in the previous chapters.

For $s = 1$, (8.14) yields

$$b_1 = 2a_{11}.$$

This condition is certainly satisfied by the one-stage Gaussian Runge-Kutta method (4.23), i.e., the implicit midpoint method, that is a symmetric method of order 2. This result is not surprising, since we have already given a direct proof of symmetry for the implicit midpoint method in Example 8.4.

For $s = 2$, (8.14) yields

$$a_{11} + a_{22} = a_{12} + a_{21}, \quad b_1 = b_2.$$

These conditions are satisfied by the two-stage Gaussian Runge-Kutta method (4.25), as well as by the two-stage Lobatto IIIA and Lobatto IIIB methods, presented in Sect. 4.4.3. Hence, these methods are symmetric.

Actually, the property is more general: all Gaussian Runge-Kutta methods (see Sect. 4.4.1) are symmetric. Similarly, all Lobatto IIIA and Lobatto IIIB (presented in Sect. 4.4.3) are symmetric as well. The interested reader can find a detailed proof in [192].

We finally aim to understand which is the connection between symplecticity and symmetry for RK methods. In some cases (as it happens for Gaussian RK methods),

the two notions coexist, while in other cases (think of Lobatto IIIA methods) they do not. The following result holds true.

Theorem 8.9 *For a given Runge-Kutta method (4.8) the following statements are equivalent:*

- *the method is symmetric for linear problems $y' = Ly$, with $L \in \mathbb{R}^{d\times d}$;*
- *the method is symplectic for problems of the type $y' = JCy$, where C is a symmetric matrix;*
- *the stability function $R(z)$ of the method, defined in (6.9), satisfies $R(-z)R(z) = 1$, for any $z \in \mathbb{C}$.*

Proof Applying a RK method to a linear problem $y' = Ly$ leads to the recurrence $y_{n+1} = R(hL)y_n$, where $R(hL)$ is the matrix version of the stability function (6.9) of the employed RK method, defined for linear scalar test problems. Symmetry holds true if and only if $y_n = R(-hL)y_{n+1}$, leading to $R(-hL)R(hL) = I$, being $I \in \mathbb{R}^{d\times d}$ is the identity matrix.

Applying a RK method to the problem $y' = JCy$ leads to $y_{n+1} = R(hJC)y_n$. As a consequence, since $\varphi_h'(y_n) = R(hJC)$, the symplecticity condition reads

$$R(hJC)^\intercal J R(hJC) = J \tag{8.15}$$

and, since for implicit Runge-Kutta methods $R(z)$ is a rational function, its matrix counterpart can be factored out as

$$R(hJC) = P(hJC)Q(hJC)^{-1}.$$

Consequently, condition (8.15) is equivalent to

$$Q(hJC)^{-\intercal} P(hJC)^\intercal J P(hJC) Q(hJC)^{-1} = J,$$

i.e.,

$$P(hJC)^\intercal J P(hJC) = Q(hJC)^\intercal J Q(hJC).$$

Algebraic manipulations of the last expression (left to the reader, see Exercise 3 at the end of this chapter) lead to $R(-hJC)R(hJC) = I$. □

Let us observe that symmetry and symplecticity are equivalent concepts if the problem is of type $y' = JCy$. This is certainly true for Hamiltonian problems with quadratic Hamiltonian function $\mathcal{H}(y) = \frac{1}{2}y^\intercal Cy$, where C is a symmetric matrix, since $\nabla\mathcal{H}(y) = Cy$.

8.6 Backward Error Analysis

As highlighted at the beginning of this chapter, a geometric numerical method is able to retain characteristic features of a dynamical system over long times. Studying the long-term character of numerical methods for ODEs has already regarded, for instance, the analysis of their linear and nonlinear stability properties, presented in the previous sections. A very effective tool in order to investigate the long-term conservative property of candidate geometric numerical methods is the *backward error analysis*, extensively presented in [192] and references therein, whose origin comes from numerical linear algebra (in particular the work of Wilkinson [345]).

The main ingredient of backward error analysis consists in inspecting the properties of differential equations associated to a numerical method, well known as *modified differential equations*, whose role is clarified in the following section.

8.6.1 Modified Differential Equations

Let us focus on the solution of an autonomous problem (1.17) by a one-step method that, over a single step, is briefly denoted as the map

$$y_n = \varphi_h(y_{n-1}).$$

Forward error analysis is performed after computing the numerical solution, by estimating the local error (i.e., the local on a single step, such as $y_1 - \Phi_h(y_0)$, being Φ the flow map of the continuous problem) or the global error (i.e., the error overall the integration interval so far, without localizing assumptions, given by $y_n - \Phi_{t_0+nh}(y_0)$).

Backward error analysis is the analysis of a continuous problem relying on the so-called modified differential equations, whose exact solution is the numerical solution of the original ODEs. More specifically, we search for an ordinary differential equation $\tilde{y}' = f_h(\tilde{y})$, written in terms of a formal power series of h, i.e.,

$$\tilde{y}' = f(\tilde{y}) + hf_2(\tilde{y}) + h^2 f_3(\tilde{y}) + \dots, \tag{8.16}$$

such that $y_n = \tilde{y}(t_0 + nh)$. The error is then measured as difference between the vector field $f(y)$ of the original problem (1.17) and that of the modified differential equation (8.16), namely $f_h(y)$. In other terms, the idea is to interpret the numerical solution computed by a given numerical method as the exact solution of a continuous problem. The right-hand side in (8.16) may generally give rise to a divergent series, so we will later employ just a truncation of it.

Under suitable regularity assumptions, the computation of modified differential equations can be provided, for instance, by means of Taylor series arguments and

using the expressions of the elementary differentials introduced in Sect. 4.2.2, as follows. Let us first expand $\tilde{y}(t+h)$ around t, leading to

$$\begin{aligned}
\tilde{y}(t+h) &= \tilde{y}(t) + h\tilde{y}'(t) + \frac{h^2}{2}\tilde{y}''(t) + \frac{h^3}{6}\tilde{y}'''(t) + \dots \\
&= \tilde{y}(t) + h\left(f + hf_2 + h^2 f_3 + \dots\right) + \frac{h^2}{2}\left(f'\tilde{y}'(t) + hf_2'\tilde{y}'(t) + \dots\right) \\
&\qquad + \frac{h^3}{6}\left(f''(f,f) + f'f'f + \dots\right) + \dots \\
&= \tilde{y}(t) + h\left(f + hf_2 + h^2 f_3 + \dots\right) \\
&\qquad + \frac{h^2}{2}\left(f' + hf_2' + \dots\right)\left(f + hf_2 + \dots\right) \\
&\qquad + \frac{h^3}{6}\left(f''(f,f) + f'f'f + \dots\right) + \dots
\end{aligned} \tag{8.17}$$

or, equivalently,

$$\begin{aligned}
\tilde{y}(t+h) = \tilde{y}(t) + hf + h^2\left(f_2 + \frac{1}{2}f'f\right) \\
+ h^3\left(f_3 + \frac{1}{2}\left(f'f_2 + f_2'f\right) + \frac{1}{6}\left(f''(f,f) + f'f'f\right)\right) + \dots
\end{aligned} \tag{8.18}$$

In the expressions above we have omitted the dependence of f, f_2, f_3 and their derivatives on $\tilde{y}(t)$, in order to simplify the notation.

Supposing that the one-step map $\phi_h(y)$ can be expanded itself in power series of h, with coefficient $f(y)$ for the power 1 due to the consistency of the method, i.e.,

$$\varphi_h(y) = y + hf(y) + h^2 d_2(y) + h^3 d_3(y) + \dots \tag{8.19}$$

yields

$$\begin{aligned}
f_2 &= d_2(y) - \frac{1}{2}f'f, \\
f_3 &= d_3(y) - \frac{1}{6}\left(f''(f,f) + f'f'f\right) - \frac{1}{2}\left(f'f_2 + f_2'f\right),
\end{aligned} \tag{8.20}$$

and so on, by comparison of (8.18) and (8.19).

Let us provide an example of computation of modified differential equations for selected numerical methods aimed to solve a scalar problem.

Example 8.6 Let us consider the following differential equation

$$y'(t) = y(t)^4, \tag{8.21}$$

assuming $y(0) = 1$ as initial value, the exact solution is

$$y(t) = \sqrt[3]{\frac{1}{1-3t}}.$$

We aim to compute the modified differential equation associated to the explicit Euler method (2.19). Clearly, in this case we have $d_j(y) = 0$ for all $j \geq 2$ in (8.19). The coefficients given in (8.20) assume the form

$$f_2(y) = -\frac{3}{2}y^5, \quad f_3(y) = \frac{19}{3}y^{10}.$$

As a consequence, the modified differential equation for the explicit Euler method applied to the logistic equation (8.21) reads

$$\tilde{y}' = \tilde{y}^4 - \frac{3}{2}h\tilde{y}^5 + \frac{19}{3}h^2\tilde{y}^{10} + \dots \tag{8.22}$$

Figure 8.11 compares the solution of the original problem based on the ODE (8.21) with the solution of the modified differential equations truncated after the h and h^2 terms. We observe that taking more terms in the modified differential equation improves the agreement between numerical and exact solutions.

The following theorem highlights an important, though expectable, property: the perturbation term in the modified differential equation of an order p method has magnitude $O(h^p)$.

Theorem 8.10 *The modified differential equation (8.16) of a one-step method $y_{n+1} = \varphi_h(y_n)$ of order p has the form*

$$\tilde{y}' = f(\tilde{y}) + h^p f_{p+1}(\tilde{y}) + h^{p+1} f_{p+2}(\tilde{y}) + \dots,$$

with $f_{p+1}(y)$ equal to the principal error term of the method.

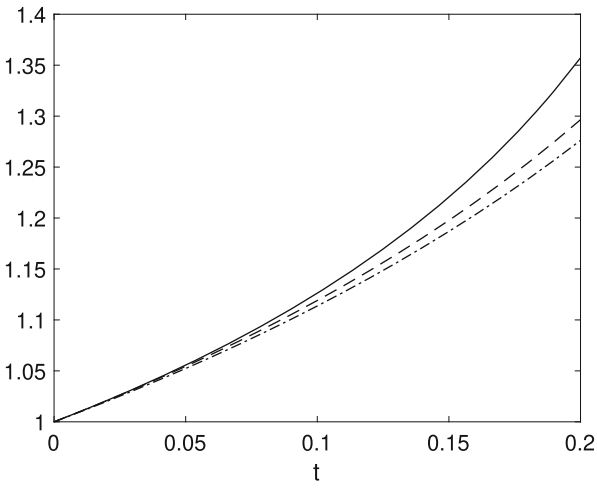


Fig. 8.11 Exact solution of Eq. (8.21) (solid line) vs solutions of the modified differential equation (8.22) of the explicit Euler method, truncated at the $O(h)$ (dashed-dotted line) and $O(h^2)$ (dashed line) terms

Proof The proof follows straightforwardly from the fact that $f_j(y) = 0$, for $2 \leq j \leq p$, if and only if $\varphi_h(y) - \Phi_h(y) = O(h^{p+1})$. □

A special case worth being considered regards the analysis of modified differential equations of symplectic methods [23, 192, 277, 336], hence with a focus on Hamiltonian problems (1.28). To this purpose, it is useful introducing the following lemma [192].

Lemma 8.2 *Let Ω be an open set of $\mathbb{R}^d$ and $f : \Omega \to \mathbb{R}^d$ be a continuously differentiable function, whose Jacobian is symmetric. Then, for any $y_0 \in \Omega$ there exists a neighborhood of y_0 and a function $\mathcal{H}(y)$ such that $f(y) = \nabla\mathcal{H}(y)$ on this neighborhood.*

Theorem 8.11 *Consider a symplectic method $\varphi_h(y)$ applied to a Hamiltonian system (1.28) with smooth Hamiltonian. Then, the corresponding modified differential equation*

$$\dot{\tilde{y}} = f(\tilde{y}) + hf_2(\tilde{y}) + h^2 f_3(\tilde{y}) + \dots$$

is also Hamiltonian. In particular, there exist smooth functions $\mathcal{H}_j : \mathbb{R}^{2d} \to \mathbb{R}$ for $j = 2, 3, \dots$, such that $f_j(y) = J\nabla\mathcal{H}_j(y)$.

Proof The proof is given by induction. In particular, since $f_1(y) = f(y) = J\nabla\mathcal{H}(y)$, we assume that $f_j(y) = J\nabla\mathcal{H}_j(y)$ is satisfied for $j = 1, 2, \ldots, r$ and aim to prove the existence of a Hamiltonian $\mathcal{H}_{r+1}(y)$. According to the inductive hypothesis, the truncated modified differential equation

$$\dot{\tilde{y}} = f(\tilde{y}) + hf_2(\tilde{y}) + \cdots + h^{r-1} f_r(\tilde{y})$$

is Hamiltonian, with Hamiltonian function given by $\mathcal{H}(y) + h\mathcal{H}_2(y) + \cdots + h^{r-1}\mathcal{H}_r(y)$. Defining its flow by $\Phi_{r,t}(y_0)$, we have

$$\begin{aligned}
\varphi_h(y_0) &= \Phi_{r,t}(y_0) + h^{r+1} f_{r+1}(y_0) + O(h^{r+2}),\\
\varphi_h'(y_0) &= \Phi_{r,t}'(y_0) + h^{r+1} f_{r+1}'(y_0) + O(h^{r+2}) \ .
\end{aligned}$$

Since the method is symplectic and the inductive hypothesis holds true, both φ_h and $\Phi_{r,h}$ are symplectic maps. Taking into account that $\Phi_{r,h}'(y_0) = I + O(h)$, we have that

$$\begin{aligned}
J &= \varphi_h'(y_0)^\intercal J \varphi_h'(y_0)\\
&= \left(\Phi_{r,t}'(y_0) + h^{r+1} f_{r+1}'(y_0)\right)^\intercal J \left(\Phi_{r,t}'(y_0) + h^{r+1} f_{r+1}'(y_0)\right) + O(h^{r+2})\\
&= \left(I + h^{r+1} f_{r+1}'(y_0)\right)^\intercal J \left(I + h^{r+1} f_{r+1}'(y_0)\right) + O(h^{r+2})\\
&= J + h^{r+1}\left(f_{r+1}'(y_0)^\intercal J + J f_{r+1}'(y_0)\right) + O(h^{r+2}).
\end{aligned}$$

This means that the matrix $J^\intercal f_{r+1}'(y_0)$ is symmetric and, by means of Lemma 8.2, there exists $\mathcal{H}_{r+1}(y)$ such that

$$J^\intercal f_{r+1}(y_0) = \nabla\mathcal{H}_{r+1}(y)$$

or, equivalently,

$$f_{r+1}(y_0) = J\nabla\mathcal{H}_{r+1}(y),$$

that completes the proof. □

We complete this section presenting a couple of results regarding the construction of the modified differential equation for the adjoint of a numerical method and, as a consequence, we provide an important result concerning the modified differential equations of symmetric methods.

Theorem 8.12 *Considering a one-step method $\varphi_h(y)$, whose modified differential equation (8.19) has coefficients $f_j(y)$, the coefficients of the modified equations of its adjoint $\varphi_h^\star(y)$ satisfy*

$$f_j^\star(y) = (-1)^{j+1} f_j(y).$$

Proof The thesis holds true in straightforward way, by considering that $\tilde{y}(t-h) = \varphi_{-h}(\tilde{y}(t))$. Consequently, it is enough to replace h by $-h$ in formulae (8.16), (8.17) and (8.19) to obtain the thesis. □

Corollary 8.1 *The right-hand side of the modified differential equation of a symmetric method only consists in even powers of h.*

Proof The thesis is direct consequence of Theorem 8.12, since a symmetric method coincides with its adjoint and, therefore, the same happens to their modified differential equations. Thus, any $f_j(y)$ is null, whenever j is even; coefficients of (8.16) with even subindices are those related to odd powers of h that, consequently, disappear from (8.16) if the method is symmetric. □

8.6.2 Truncated Modified Differential Equations

As aforementioned, the presentation of modified differential equations so far has been based on considering their right-hand side as a formal series of powers of h, without taking into account its convergence. Unfortunately, as clearly highlighted in [192], such a power series is almost never convergent, actually even in very simple situations. As a consequence, we should consider a proper truncation of the modified differential equations, up to an optimal index to be properly chosen. Such a choice is based on rigorous error estimates, described in details in [192] and references therein. Here we report them without their proofs, that can be found in the mentioned monograph by Hairer, Lubich and Wanner.

We aim to find an optimal truncation index N for the modified differential equation (8.16) leading to

$$\tilde{y}' = F_N(\tilde{y}) = f(\tilde{y}) + hf_2(\tilde{y}) + \ldots + h^{N-1} f_N(\tilde{y}),$$

with $\tilde{y}(0) = y_0$. To this purpose, the following bound on the coefficients of (8.16), whose proof can be found in [192], is particularly useful.

Theorem 8.13 *Suppose that $f(y)$ is analytic in $\mathcal{B}_{2R}(y_0)$ and the coefficients of (8.19) are also analytic in $\mathcal{B}_R(y_0)$. Assume that there exists a positive M such that $\|f(y)\| \leq M$, for any $\|y - y_0\| \leq 2R$. Moreover, assume that each $d_j(y)$ in (8.19) satisfies*

$$\|d_j(y)\| \leq \mu M \left(\frac{2\kappa M}{R}\right)^{j-1},$$

for any $\|y - y_0\| \leq R$, where

$$\mu = \sum_{i=1}^{s} |b_i|, \quad \kappa = \max_{i=1,2,\ldots,s} \sum_{j=1}^{s} |a_{ij}|.$$

Then, the following bound holds true

$$\|f_j(y)\| \leq \ln 2 \, \eta M \left(\frac{\eta M j}{R}\right)^{j-1}, \tag{8.23}$$

assuming that $\|y - y_0\| \leq R/2$ and being $\eta = 2\max(\kappa, \mu/(2\ln 2 - 1))$.

Taking into account the bound (8.23) and since the function $(\varepsilon x)^x$ has a minimum at $x = (\varepsilon \mathrm{e})^{-1}$, it makes sense assuming as truncation index the integer N such that

$$\frac{\eta M N}{R} \leq \frac{1}{h\mathrm{e}}$$

or, in less restrictive way,

$$hN \leq \mathrm{e}h_0,$$

being $h_0 = \frac{R}{\mathrm{e}\eta M}$. In this way, since $\|f(y)\| \leq M$ and using (8.23), we have

$$\|F_N(y)\| \leq M \left(1 + \eta \ln 2 \sum_{j=2}^{N} \left(\frac{\eta M j}{R}\right)^{j-1}\right) \leq M \left(1 + \eta \ln 2 \sum_{j=2}^{N} \left(\frac{j}{hN}\right)^{j-1}\right),$$

leading to

$$\|F_N(y)\| \leq M(1 + 1.65\eta).$$

The following result holds true (see [192]).

Theorem 8.14 *Let $f(y)$ be analytic in $\mathcal{B}_{2R}(y_0)$ and the coefficients $d_j(y)$ of (8.19) analytic in $\mathcal{B}_R(y_0)$. If $h \leq h_0/4$, then there exists $N = N(h)$ (the largest integer satisfying $hN \leq h_0$), such that*

$$\|\varphi_h(y_0) - \Phi_{N,h}(y_0)\| \leq h\gamma M e^{-h_0/h},$$

with $\gamma = \mathrm{e}(2 + 1.65 + \mu)$ only depending on the method.

In other terms, for problems with analytic vector fields, the numerical solution computed by a one-step method and the solution of the corresponding modified differential equation, truncated after $N \sim \frac{1}{h}$ terms, differ by a term that is exponentially small.

8.6.3 Long-Term Analysis of Symplectic Methods

The core of backward error analysis in the context of geometric numerical integration certainly involves the study of the long-time conservative character of symplectic numerical methods applied to Hamiltonian problems (1.28). We know from Theorem 8.11 that the corresponding modified differential equation is also Hamiltonian and, after truncation, the modified Hamiltonian is given by

$$\widetilde{\mathcal{H}}(y) = \mathcal{H}(y) + h^p \mathcal{H}_{p+1}(y) + \cdots + h^{N-1}\mathcal{H}_N(y). \tag{8.24}$$

The following fundamental result, proved by Benettin and Giorgilli in [23], provides information on the long-term conservative character of symplectic methods.

Theorem 8.15 (Benettin-Giorgilli Theorem) *Consider a Hamiltonian system (1.28) with analytic Hamiltonian function $\mathcal{H} : D \to \mathbb{R}$, with $D \subset \mathbb{R}^{2d}$. Suppose that a symplectic numerical method $\varphi_h(y)$ of order p is used to solve this problem and assume that the corresponding numerical solution lies in a compact set $K \subset D$. Then, there exists h_0 and $N = N(h)$ (as in Theorem 8.13)*

(continued)

Theorem 8.15 (continued)
such that

$$\begin{aligned}\widetilde{\mathcal{H}}(y_n) &= \widetilde{\mathcal{H}}(y_0) + \mathcal{O}(e^{-h_0/2h}),\\ \mathcal{H}(y_n) &= \mathcal{H}(y_0) + \mathcal{O}(h^p),\end{aligned} \tag{8.25}$$

for exponentially long time intervals of length $nh - t_0 \leq e^{h_0/2h}$.

Proof Let $\Phi_{N,t}(y_0)$ be the flow of the truncated modified equation (8.24), that is also Hamiltonian with Hamiltonian function $\widetilde{\mathcal{H}}$ satisfying $\widetilde{\mathcal{H}}\left(\Phi_{N,t}(y_0)\right) = \widetilde{\mathcal{H}}(y_0)$, for any t. As a consequence of Theorem 8.14, we have that

$$\|y_{n+1} - \Phi_{N,h}(y_n)\| \leq h\gamma M e^{-h_0/h}$$

and again, from Theorem 8.13, we deduce that there exists a global Lipschitz constant (independent from h) for $\widetilde{\mathcal{H}}$, such that

$$\widetilde{\mathcal{H}}(y_n) - \widetilde{\mathcal{H}}(\Phi_{N,h}(y_n)) = \mathcal{O}(he^{-h_0/h}) \ .$$

Since

$$\widetilde{\mathcal{H}}(y_n) - \widetilde{\mathcal{H}}(y_0) = \sum_{j=1}^{n}\left(\widetilde{\mathcal{H}}(y_j) - \widetilde{\mathcal{H}}(y_{j-1})\right) = \sum_{j=1}^{n}\left(\widetilde{\mathcal{H}}(y_j) - \widetilde{\mathcal{H}}(\Phi_{N,h}(y_{j-1}))\right) \ ,$$

we obtain $\widetilde{\mathcal{H}}(y_n) - \widetilde{\mathcal{H}}(y_0) = \mathcal{O}(nhe^{-h_0/h})$, that proves the statement for $\widetilde{\mathcal{H}}$, recalling that $nh \leq e^{h_0/2h}$.

The result for $\mathcal{H}$ follows from (8.24), since

$$\begin{aligned}\widetilde{\mathcal{H}}(y) &= \mathcal{H}(y) + h^p\mathcal{H}_{p+1}(y) + \cdots + h^{N-1}\mathcal{H}_N(y)\\ &= \mathcal{H}(y) + h^p\left(\mathcal{H}_{p+1}(y) + h\mathcal{H}_{p+2}(y) + \cdots + h^{N-p-1}\mathcal{H}_N(y)\right)\end{aligned}$$

and considering the fact that

$$\mathcal{H}_{p+1}(y) + h\mathcal{H}_{p+2}(y) + \cdots + h^{N-p-1}\mathcal{H}_N(y)$$

is uniformly bounded on K, independently of h and N. This is a consequence of the fact that

$$\mathcal{H}_j(y) = \int_0^1 y^{\intercal} f_j(ty)\mathrm{d}t + \text{constant}$$

on a ball centered in y_0 contained in D and, moreover, of the estimate on f_j given by (8.23). □

Benettin-Giorgilli theorem 8.15 is a gifted result in understanding the long-term conservative character of a symplectic method: as long as the numerical solution lies in a compact set, the Hamiltonian function of the optimally truncated modified differential equation is almost conserved up to errors of exponentially small size. Moreover, for a symplectic method of order p, the modified Hamiltonian function is close to the original Hamiltonian function over exponentially long time windows, with a deviation comparable to the accuracy in the computation of the solution, i.e., $O(h^p)$. Let us test the usefulness of this result through the following highly didactic example.

Example 8.7 Let us apply Benettin-Giorgilli theorem to the mathematical pendulum (1.23), with $p_0 = 0$ and $q_0 = 1$. The reader can find a detailed verification of the hypothesis of Theorem 8.15 for this problem in [192] (Example VI.8.2). Actually, the stepsize restriction dictated by Theorem 8.14 is too severe and definitely not sharp. Indeed, symplectic methods may have excellent conservation properties even if used with large values of the stepsize.

We use the symplectic Euler method (8.9) and the two-stage Gaussian method (4.25) with several values of the stepsize. As visible in Fig. 8.12, the conservation of the symplectic structure is achieved also for large values of h.

Let us now check the accuracy in conserving the Hamiltonian function. Figures 8.13 and 8.14 reveal an excellent long-term conservation of the Hamiltonian, measured for several values of the stepsize, in the intervals [0,1000] and [0,10000]. The accuracy of the second equation in (8.25) is also confirmed, as visible in Tables 8.1 and 8.2, where the orders of both methods are very well recovered. They have been computed through the following formula, analogous to (3.23),

$$p \approx \log_2 \left| \frac{\mathcal{H}(y_N) - \mathcal{H}(y_0)}{\mathcal{H}(y_{2N}) - \mathcal{H}(y_0)} \right|, \tag{8.26}$$

i.e., as the logarithm in basis 2 of the ratio of the deviations between the Hamiltonian in the numerical solution computed with stepsize h from the initial Hamiltonian, divided by the analogous deviation with stepsize $h/2$. Both values are listed in the table with reference to the final integration point.

Let us finally make an observation on non-symplectic methods, motivated by Fig. 8.4, where a linear energy drift is visible for the explicit Euler method. This fact can be motivated through arguments very similar to those provided in the proof of Benettin and Giorgilli theorem (8.15). Indeed, one can prove (also see Exercise 6

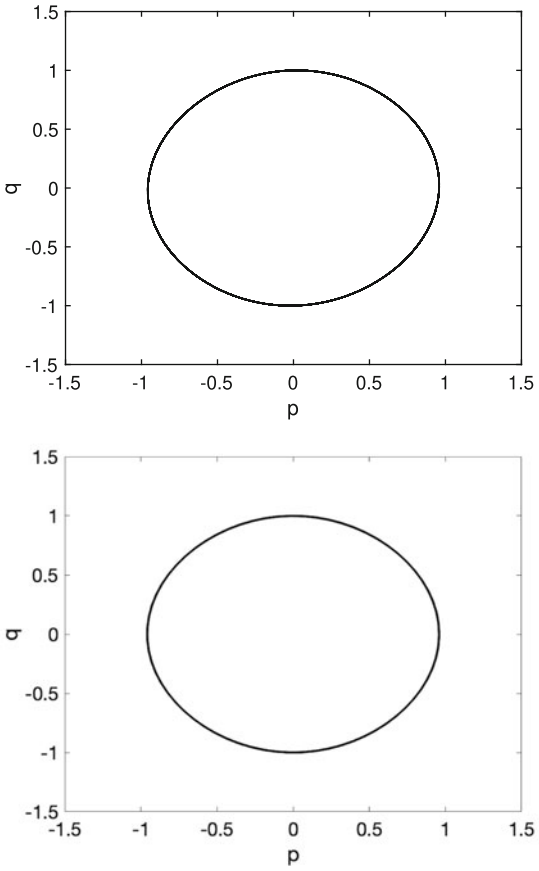


Fig. 8.12 Example 8.7: phase portrait associated to the numerical dynamics generated by applying the symplectic Euler method (8.9) (top) and the two-stage Gaussian method (4.25) (bottom) to the mathematical pendulum (1.23). The graphs are obtained in correspondence of $h = 0.05$ (top) and $h = 0.1$ (bottom)

at the end of the chapter) that

$$\mathcal{H}(y_n) = \mathcal{H}(y_0) + O(th^p).$$

We finally observe that alternatives to symplecticity or relaxed notions of symplecticity have been treated in the literature, e.g., through the notion of conjugate symplectic method [133, 192, 197].

8.7 Long-Term Analysis of Multivalue Methods

This section is devoted to providing a comprehensive analysis of the long-term stability properties of multivalue numerical methods, described in Chap. 5. The presented analysis is based on the results contained in [122].

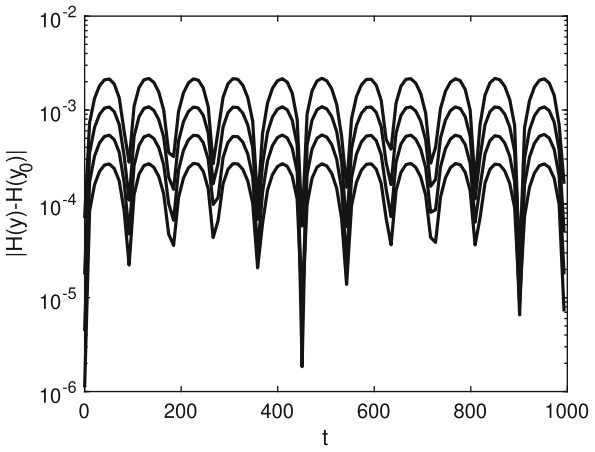


Fig. 8.13 Example 8.7: Hamiltonian deviations along the numerical dynamics generated by applying the symplectic Euler method (8.9) to the mathematical pendulum (1.23). The graphs are obtained in correspondence of four values of the stepsize: $h = 0.01$ (top), $h = 0.005, 0.0025$ (middle) and $h = 0.00125$ (bottom). The plot displays the graph obtained considering a grid point every hundred

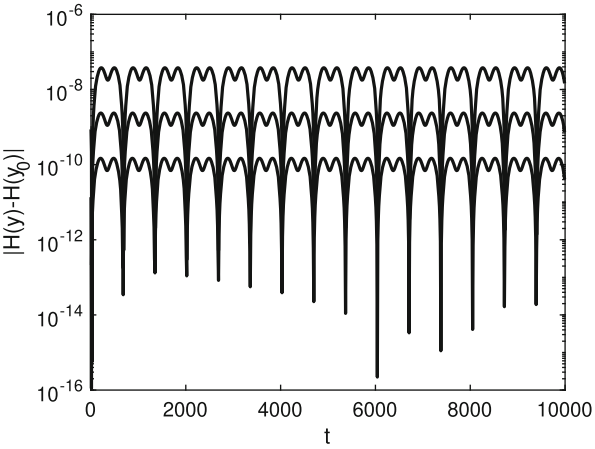


Fig. 8.14 Example 8.7: Hamiltonian deviations along the numerical dynamics generated by applying the two-stage Gaussian method (4.25) to the mathematical pendulum (1.23). The three graphs are obtained in correspondence of three values of the stepsize: $h = 0.1$ (top), $h = 0.05$ (middle) and $h = 0.025$ (bottom). The plot displays the graph obtained considering a grid point every hundred

Table 8.1 Example 8.7: Hamiltonian deviations along the numerical dynamics generated by applying the symplectic Euler method (8.9) to the mathematical pendulum (1.23), computed in the final integration point $t = 1000$. The displayed Hamiltonian deviations measure the gap at the final step point from the initial Hamiltonian. Order estimation is also reported, computed as suggested by Eq. (8.26)

h	Hamiltonian deviation (final point)	p
0.01	$1.64 \cdot 10^{-4}$	
0.005	$7.27 \cdot 10^{-5}$	1.17
0.0025	$3.49 \cdot 10^{-5}$	1.06
0.00125	$1.69 \cdot 10^{-5}$	1.05

Table 8.2 Example 8.7: Hamiltonian deviations along the numerical dynamics generated by applying the two-stage Gaussian method (4.25) to the mathematical pendulum (1.23), computed in the final integration point $t = 10000$. The displayed Hamiltonian deviations measure the gap at the final step point from the initial Hamiltonian. Order estimation is also reported, computed as suggested by Eq. (8.26)

h	Hamiltonian deviation (final point)	p
0.05	$6.74 \cdot 10^{-9}$	
0.025	$4.23 \cdot 10^{-10}$	3.99
0.0125	$2.64 \cdot 10^{-11}$	4.00

To perform the long-term analysis of multivalue methods, it is worth using the following representation for the forward step procedure

$$Y_{n+1} = V\, Y_n + h\, \Phi(h, Y_n). \tag{8.27}$$

We also remind that the method requires a starting procedure

$$Y_0 = \mathcal{S}_h(y_0),$$

and a finishing procedure

$$y_n = \mathcal{F}_h(Y_n),$$

which permits to extract the numerical approximation from Y_n. If d is the dimension of the differential equation (1.17) and V is a matrix of dimension $r \times r$ (by abuse of notation we write V in (8.27) instead of the correct $V \otimes I$, where I is the d-dimensional identity matrix), then the vector Y_n is of dimension rd.

If $r > 1$, the recursion of the forward step procedure has parasitic solutions. Our aim is to study the long-time behavior of these parasitic solutions. We are mainly interested in stable methods having good conservation properties. We therefore assume that all eigenvalues of V are simple and lie on the unit circle. We denote them by $\zeta_1 = 1, \zeta_2, \ldots, \zeta_r$. We let v_j and v_j^* be right and left eigenvectors ($V v_j = \zeta_j v_j$ and $v_j^* V = \zeta_j v_j^*$) satisfying $v_j^* v_j = 1$.

To relate the forward step procedure (8.27) to the differential equation (1.17) we assume the pre-consistency condition

$$\Phi(0, Y) = Bf(UY), \qquad Uv_1 = e, \tag{8.28}$$

where B is an $r \times s$ matrix, U an $s \times r$ matrix, and e is the unit vector in $\mathbb{R}^s$. Again, by abuse of notation, we avoid the heavy tensor notation and use matrices B and U instead of $B \otimes I$ and $U \otimes I$. For $UY = W = (W_i)_{i=1}^s \in \mathbb{R}^{sd}$ the vector $f(W) \in \mathbb{R}^{sd}$ is defined by $f(W) = (f(W_i))_{i=1}^s$. We assume throughout this article that the forward step method is consistent, i.e.,

$$v_1^* \Phi(0, yv_1) = f(y), \tag{8.29}$$

and, for pre-consistent methods (8.28), it is equivalent to $v_1^* Be = 1$.

8.7.1 Modified Differential Equations

As discussed for one-step methods, a crucial tool for the study of the long-time behavior of numerical integrators is the backward error analysis, extended to the case of multivalue methods in [122]. This analysis relies on describing the dynamics of the smooth and parasitic components characterizing the numerical solution computed by genuine multivalue methods (i.e., those with $r > 1$).

With the aim of separating the smooth and parasitic components in the numerical solution $y_n = \mathcal{F}_h(Y_n)$, we consider approximations to Y_n of the form

$$\widehat{Y}_n = Y(t_n) + \sum_{j=2}^{r} \zeta_j^n \, Z_j(t_n), \tag{8.30}$$

where $t_n = nh$, and the coefficient functions $Y(t)$, $Z_j(t)$ are independent of n, but depend smoothly on h. Such expansions have first been considered for the study of the long-time behavior of linear multistep methods [187] (also refer to [191, 193] for highly oscillatory problems).

We introduce a system of modified differential equations for the smooth functions $Y(t)$ and $Z_j(t)$. These modified equations only depend on the forward step procedure and are independent of the starting and finishing procedures.

Theorem 8.16 *Consider a forward step procedure (8.27) with matrix V having simple eigenvalues of modulus* 1. *Then, there exist h-independent real functions $f_l(y_1)$ and complex functions $g_{kl}(y_1), a_{jl}(y_1)$ and $b_{jkl}(y_1)$ such*

(continued)

Theorem 8.16 (continued)
that, for an arbitrarily chosen truncation index N and for any solution $y_k(t)$, $z_{kj}(t)$, $j,k = 1,2,\ldots,r$, of the system

$$\begin{aligned}
\dot{y}_1 &= f(y_1) + h\, f_1(y_1) + \ldots + h^{N-1} f_{N-1}(y_1),\\
y_k &= h\, g_{k1}(y_1) + \ldots + h^N g_{k,N}(y_1), \qquad k > 1,\\
\dot{z}_{jj} &= \big(a_{j0}(y_1) + h\, a_{j1}(y_1) + \ldots + h^{N-1} a_{j,N-1}(y_1)\big) z_{jj},\\
z_{jk} &= \big(h\, b_{jk1}(y_1) + \ldots + h^N b_{j,k,N}(y_1)\big) z_{jj}, \qquad k \neq j,
\end{aligned} \tag{8.31}$$

the approximations (8.30), with

$$Y(t) = \sum_{k=1}^{r} y_k(t)\, v_k, \qquad Z_j(t) = \sum_{k=1}^{r} z_{kj}(t)\, v_k, \tag{8.32}$$

satisfy (8.27) with a small defect, i.e.,

$$\widehat{Y}_{n+1} = V\, \widehat{Y}_n + h\, \Phi(h, \widehat{Y}_n) + O(h^{N+1}), + O(h\|\mathbf{Z}\|^2),$$

as long as $y_1(t_n)$ remains in a compact set. The constant symbolized by $O(\cdot)$ is independent of h, but depends on the truncation index N. We use the notation $\|\mathbf{Z}\| = \max\{|z_{jk}(t_n)|\,;\ j,k = 1,\ldots,r\}$.

Proof Inserting (8.30) into the forward step procedure and expanding the nonlinearity around $Y(t_n)$ yields

$$\begin{aligned}
Y(t+h) &= V\, Y(t) + h\, \Phi\big(h, Y(t)\big) + O(h\|\mathbf{Z}\|^2)\\
\zeta_j\, Z_j(t+h) &= V\, Z_j(t) + h\, \Phi'\big(h, Y(t)\big)\, Z_j(t) + O(h\|\mathbf{Z}\|^2).
\end{aligned} \tag{8.33}$$

Neglecting terms of size $O(h\|\mathbf{Z}\|^2)$ and using (8.32), from the previous relation we get

$$y_k(t+h) = \zeta_k\, y_k(t) + h\, v_k^* \Phi\big(h, Y(t)\big).$$

We expand the left-hand side into a Taylor series around $h = 0$ and thus obtain (omitting the argument t)

$$\begin{aligned} \dot{y}_1 + \frac{h}{2}\,\ddot{y}_1 + \cdots &= \Psi_1(h, y_1, \ldots, y_r) \\ (1-\zeta_k)\,y_k + h\,\dot{y}_k + \frac{h^2}{2}\,\ddot{y}_k + \cdots &= h\Psi_k(h, y_1, \ldots, y_r), \quad k = 2, \ldots, r. \end{aligned} \tag{8.34}$$

Differentiation of the relations for y_k $(k = 2, \ldots, r)$ and recursive elimination of the first and higher derivatives, and also of $y_2, \ldots, y_r$ on the right-hand side, yield the second relation of (8.31) with a defect of size $O(h^{N+1})$. In the same way one can eliminate the second and higher derivatives in the first equation of (8.34) and thus obtains a differential equation for y_1. By the consistency assumption (8.29), the h-independent term of this differential equation becomes $f(y_1)$.

Neglecting terms of size $O(h\|\mathbf{Z}\|^2)$ in the second relation of (8.33) yields

$$\zeta_j z_{kj}(t+h) = \zeta_k z_{kj}(t) + h\, v_k^* \Phi'\big(h, Y(t)\big)\, Z_j(t). \tag{8.35}$$

We expand the left-hand side into a Taylor series, and apply the same elimination procedure as for the smooth component $Y(t)$. This then gives a first order differential equation for z_{jj} and algebraic relations for z_{kj} $(k \neq j)$, and terminates the proof of (8.31). □

It is now worth equipping modified differential equations by suitable initial conditions. For $n = 0$ and $\widehat{Y}_0 = Y_0 = \mathcal{S}_h(y_0)$ the relation (8.30) gives

$$\mathcal{S}_h(y_0) = Y(0) + \sum_{j=2}^{r} Z_j(0).$$

Because of the algebraic relations in (8.31), this represents a nonlinear algebraic equation for the h-dependent vectors $y_1(0), z_{22}(0), \ldots, z_{rr}(0)$. For $h = 0$, we get

$$y_1(0)\big|_{h=0} = v_1^* \mathcal{S}_0(y_0), \qquad z_{jj}(0)\big|_{h=0} = v_j^* \mathcal{S}_0(y_0),$$

and the implicit function theorem guarantees the existence of a local unique solution for sufficiently small h.

The initial values $z_{jj}(0)$, for $j = 2, \ldots, r$, determine, on intervals of length $O(1)$, the size of the parasitic solution components. We shall investigate how they depend on the choice of the starting procedure. Let us denote the forward step procedure (8.27) by $Y_{n+1} = \mathcal{G}_h(Y_n)$. We know from Sect. 5.1 (also see Theorem XV.8.2 of [192]) that, for a given $\mathcal{G}_h(Y)$ and a given finishing procedure $\mathcal{F}_h(Y)$, there exist a unique (as formal power series in h) starting procedure $\mathcal{S}_h^*(y)$ and a unique one-step

method $y_{n+1} = \Phi_h^*(y_n)$, such that

$$\mathcal{G}_h \circ \mathcal{S}_h^* = \mathcal{S}_h^* \circ \Phi_h^* \qquad \text{and} \qquad \mathcal{F}_h \circ \mathcal{S}_h^* = \text{identity}. \tag{8.36}$$

This means that for the choice $Y_0 = \mathcal{S}_h^*(y_0)$ the numerical solution obtained by the multivalue method is (formally) equal to that of the one-step method Φ_h^*, the so-called underlying one-step method.

For all common multivalue methods, the underlying one-step method and the components of the starting procedure are B-series. Their coefficients can be computed recursively from the relations (8.36) by using the composition formula for B-series.

Theorem 8.17 *Let the starting procedure $\mathcal{S}_h(y_0)$ satisfy*

$$\mathcal{S}_h(y_0) = \mathcal{S}_h^*(y_0) + O(h^q), \tag{8.37}$$

and assume that the finishing procedure is given by $F_h(Y) = v_1^* Y = y_1$. Then, the initial values for the system of modified equations (8.31) satisfy

$$y_1(0) = y_0 + O(h^q), \qquad z_{jj}(0) = O(h^q).$$

Proof For the exact starting procedure $\mathcal{S}_h^*(y_0)$, the numerical solution $\{y_n\}_{n\geq 0}$ is that of the underlying one-step method and does not have parasitic components. Consequently, we have $y_1(0) = y_0$ and $z_{kj}(0) = 0$ for all k and j. A perturbation of this starting procedure implies, by the implicit function theorem, a perturbation of the same size in the initial values $y_1(0), z_{22}(0), \ldots, z_{rr}(0)$. □

We conclude this section by providing a result regarding the modified differential equations of symmetric multivalue methods, according to the following definition of symmetry.

Definition 8.6 A given multivalue method (8.27) is *symmetric* if its underlying one-step method is a symmetric method.

Theorem 8.18 *Consider a forward step procedure (8.27), where V is of dimension 2 with eigenvalues 1 and -1, and assume that the method is*

(continued)

Theorem 8.18 (continued)
symmetric, therefore mathematically equivalent to

$$Y_n = V\, Y_{n+1} - h\, \Phi(-h, Y_{n+1}).$$

Then, Eq. (8.31) only contain expressions with even powers of h.

Proof Neglecting terms of size $O(h^{N+1})$ and $O(h\|\mathbf{Z}\|^2)$, the functions $Y(t)$ and $Z_j(t)$ of Theorem 8.16 satisfy

$$\begin{aligned} Y(t+h) &= V\, Y(t) + h\, \Phi\big(h, Y(t)\big), \\ \zeta_j\, Z_j(t+h) &= V\, Z_j(t) + h\, \Phi'\big(h, Y(t)\big)\, Z_j(t), \end{aligned} \tag{8.38}$$

where the prime in $\Phi'(h, Y)$ stands for a derivative with respect to Y. Our assumption on the forward step procedure implies that

$$\begin{aligned} Y(t) &= V\, Y(t+h) - h\, \Phi\big(-h, Y(t+h)\big), \\ Z_j(t) &= V\, \zeta_j\, Z_j(t+h) - h\, \Phi'\big(-h, Y(t+h)\big)\, \zeta_j\, Z_j(t+h), \end{aligned}$$

and, replacing $t - h$ for t, leading to

$$\begin{aligned} Y(t-h) &= V\, Y(t) - h\, \Phi\big(-h, Y(t)\big), \\ \zeta_j^{-1} Z_j(t-h) &= V\, Z_j(t) - h\, \Phi'\big(-h, Y(t)\big)\, Z_j(t). \end{aligned} \tag{8.39}$$

Let us first consider the components of the vector $Y(t)$. Comparing the upper relations of (8.38) and (8.39) we notice that the components $y_k(t)$ of $Y(t)$ have to satisfy the same equations for h and for $-h$.

Since, by assumption, $\zeta_2 = -1$ is the only eigenvalue of V different from 1, we have $\zeta_2^{-1} = \zeta_2$. The lower relation of (8.38) is therefore equal to the lower relation of (8.39), where h is replaced by $-h$. Consequently, also the components of $Z_2(t)$ have to satisfy the same equations for h and for $-h$. This implies that all equations of (8.31) are in even powers of h. □

8.7.2 Bounds on the Parasitic Components

The parasitic solution components are determined by the functions $z_{jj}(t)$. To study their long-time behavior we first examine the leading term in the differential

equation (8.31) for z_{jj}. For $k = j$, Eq. (8.35) yields

$$\zeta_j \, \dot{z}_{jj} = v_j^* \Phi'(0, y_1 v_1) \, v_j z_{jj} + O(h|z_{jj}|).$$

Subject to the pre-consistency assumption (8.28), we obtain

$$\dot{z}_{jj} = \mu_j \, f'(y_1) \, z_{jj} + O(h|z_{jj}|), \qquad \mu_j = \zeta_j^{-1} \, v_j^* B U v_j. \tag{8.40}$$

The coefficients μ_j are called *growth parameters* of the multivalue method. They determine to a large extent the long-term behavior of the parasitic components $Z_j(t)$.

It follows from Theorem 8.16 that the coefficient functions of the parasitic solution components (8.32) satisfy

$$\begin{aligned} \dot{z}_{jj} &= h^M A\big(h, y_1(t)\big) \, z_{jj}, \\ z_{jk} &= h \, B\big(h, y_1(t)\big) \, z_{jj}, \qquad k \neq j. \end{aligned} \tag{8.41}$$

In general we have $M = 0$, but if the growth parameters (8.40) of the method are zero we have $M = 1$, and if in addition to zero growth parameters the assumptions of Theorem 8.18 are satisfied we have $M = 2$. If the vector field $f(y)$ of (1.17) is smooth and has bounded derivatives (which excludes stiff and highly oscillatory problems), the functions $A(h, y_1)$ and $B(h, y_1)$ are bounded as long as $y_1(t)$ stays in a compact set. Grönwall lemma then implies

$$\|z_{jj}(t)\| \leq \|z_{jj}(0)\| \, \exp(h^M L t), \tag{8.42}$$

where L is a bound on the norm or, better, the logarithmic norm of $A(h, y_1)$. For $k \neq j$ the functions $z_{jk}(t)$ are bounded by the same expression with an additional factor Ch.

8.7.3 Long-Time Conservation for Hamiltonian Systems

We have built the necessary tools to prove a conservation result for multivalue methods applied to Hamiltonian problems (1.22), as follows.

Theorem 8.19 *Consider a multivalue method of order p, a starting procedure satisfying (8.37) with q, and let $0 \leq M \leq q$ be the integer such that the modified equations for z_{jj}, $j = 2, \ldots, r$, satisfy (8.41). Furthermore, assume the existence of a modified Hamiltonian $\widetilde{\mathcal{H}}(y)$ satisfying $\widetilde{\mathcal{H}}(y) - \mathcal{H}(y) =$*

(continued)

Theorem 8.19 (continued)
$O(h^p)$ which is well preserved by the flow $\widetilde{\varphi}_t(y)$ of the underlying one-step method, more precisely,

$$\widetilde{\mathcal{H}}\big(\widetilde{\varphi}_h(y)\big) = \widetilde{\mathcal{H}}(y) + O(h^{\gamma+1}), \tag{8.43}$$

with $p \le \gamma \le 2q$. We then have, for $t = nh$,

$$\mathcal{H}(y_n) - \mathcal{H}(y_0) = O(h^p) + O(th^\gamma) + O\big(h^{q+1}\exp(h^M Lt)\big),$$

as long as $t = O(h^{-M})$.

Proof Recall that for a given initial value y_0 the numerical solution is obtained from $Y_0 = \mathcal{S}_h(y_0)$, the forward step procedure $Y_{n+1} = VY_n + h\Phi(h, Y_n)$, and the finishing procedure $y_n = \mathcal{F}_h(Y_n)$. The proof consists in several steps.

(a) We use the expansion (8.30) only locally, on one step. This means that, for any n, we compute functions $Y^{[n]}(t)$ and $Z_j^{[n]}(t)$ satisfying the modified equations (8.31), such that

$$Y_n = Y^{[n]}(0) + \sum_{j=2}^{r} Z_j^{[n]}(0).$$

It follows from Theorem 8.16 that (with the choice $N = 2q$)

$$Y_{n+1} = Y^{[n]}(h) + \sum_{j=2}^{r} \zeta_j Z_j^{[n]}(h) + O(h^{2q+1}),$$

as long as the parasitic components are bounded as $\|Z(t)\| = O(h^q)$. By the uniqueness of the initial values, we have that

$$Y^{[n+1]}(0) = Y^{[n]}(h) + O(h^{2q+1}), \qquad Z_j^{[n+1]}(0) = \zeta_j Z_j^{[n]}(h) + O(h^{2q+1}). \tag{8.44}$$

(b) The estimates (8.42) and (8.44) yield

$$\|z_{jj}^{[n+1]}(0)\| \le \|z_{jj}^{[n]}(h)\| + Ch^{2q+1} \le \|z_{jj}^{[n]}(0)\| \exp(h^{M+1}L) + Ch^{2q+1}.$$

Applying a discrete Gronwall Lemma we obtain for $t = nh$

$$\|z_{jj}^{[n]}(0)\| \le \|z_{jj}^{[0]}(0)\| \exp(h^M Lt) + Ch^{2q}\, t\, \exp(h^M Lt). \tag{8.45}$$

(c) We assume that the finishing procedure is given by $\mathcal{F}_h(Y) = v_1^* Y$, so that the flow of the modified equation for y_1 in (8.31) represents the underlying one-step method. We consider the telescoping sum

$$\widetilde{\mathcal{H}}\big(y_1^{[n]}(0)\big) - \widetilde{\mathcal{H}}\big(y_1^{[0]}(0)\big) = \sum_{l=0}^{n-1} \Big(\widetilde{\mathcal{H}}\big(y_1^{[l+1]}(0)\big) - \widetilde{\mathcal{H}}\big(y_1^{[l]}(0)\big) \Big).$$

From the estimate (8.44) and the assumption (8.43) we obtain that every summand is bounded by $O(h^{2q+1}) + O(h^{\gamma+1})$ (the first term can be removed, because $\gamma \le 2q$), which yields an error term of size $O(th^{\gamma})$. In the left-hand side we substitute $y_1^{[n]}(0)$ from the relation

$$y_n = y_1^{[n]}(0) + \sum_{j=2}^{r} z_{1j}^{[n]}(0).$$

The statement now follows from $\|z_{1j}(0)\| \le ch\|z_{jj}(0)\|$, from the bounds (8.45) for $z_{jj}^{[n]}(0)$, and from the assumption $\widetilde{\mathcal{H}}(y) - \mathcal{H}(y) = O(h^p)$.

□

The crucial ingredient of the previous theorem is the existence of a modified Hamiltonian function. Let us discuss some relevant situations where such a modified Hamiltonian is known to exist.

- If the underlying one-step method is a symplectic transformation, there exists a modified Hamiltonian satisfying (8.43) with arbitrarily large γ (see Sect. IX.3 in [192]; also see Theorem 8.15). Unfortunately, the underlying one-step method of multivalue methods cannot be symplectic [190];
- if (1.22) is an integrable reversible system, and if the underlying one-step method is symmetric (reversible), under mild non-resonance conditions there exists a modified Hamiltonian satisfying (8.43) with arbitrarily large γ (see Chapter 9 in [192]);
- if the underlying one-step method is a B-series (this is the case for all general linear methods), necessary and sufficient conditions for the existence of a modified Hamiltonian satisfying (8.43) with a given γ are presented in [192] (Chapter IX.9.4). For example, only one condition is necessary for symmetric methods of order 4 to satisfy condition (8.43) with $\gamma = 6$.

Example 8.8 Let us consider a multivalue method in the following form

$$Y_{n+1} = VY_n + hBf(W), \qquad W = UY_n + hAf(W).$$

(continued)

Example 8.8 (continued)
with

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cccc|cc} \frac{1}{12} & 0 & 0 & 0 & 1 & \frac{1}{2} \\ -\frac{1}{3} & \frac{1}{6} & 0 & 0 & 1 & 1 \\ \frac{5}{3} & -\frac{2}{3} & \frac{1}{6} & 0 & 1 & -1 \\ \frac{7}{6} & -\frac{5}{12} & \frac{1}{12} & \frac{1}{12} & 1 & -\frac{1}{2} \\ \hline \frac{2}{3} & -\frac{1}{6} & -\frac{1}{6} & \frac{2}{3} & 1 & 0 \\ 1 & -\frac{1}{2} & \frac{1}{2} & -1 & 0 & -1 \end{array}\right],$$

corresponding to a multivalue method proposed in [73] and analyzed in [122]. The vector

$$Y_n = \begin{bmatrix} y_n \\ a_n \end{bmatrix}$$

provides an approximation y_n to the solution and an approximation a_n to a scaled second derivative. If we denote by $R_h(y_0)$ the result of one step of the Runge-Kutta method

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & \frac{373}{550} & \frac{177}{550} & \\ 0 & \frac{8233}{50976} & -\frac{30749}{152928} & \frac{3025}{76464} \\ \hline & 0 & -\frac{383}{648} & \frac{275}{1296} \quad 1 \end{array}$$

then the starting procedure is given by

$$\mathcal{S}_h(y_0) = \begin{bmatrix} y_0 \\ \frac{1}{2}\big(R_h(y_0) + R_{-h}(y_0)\big) - y_0 \end{bmatrix}.$$

Let us collect some essential properties of this method:

- the method has order $p = 4$, implying that the underlying one-step method has also order 4;
- the method is symmetric in the sense of Theorem 8.18. As a consequence all equations in (8.31) are in even powers of h;

(continued)

Example 8.8 (continued)
- the eigenvalues of V are $\zeta_1 = 1$ and $\zeta_2 = -1$. By construction, the growth parameter corresponding to the parasitic root $\zeta_2 = -1$ is zero. Together with the symmetry of the method this implies that $M = 2$ in (8.41);
- the analysis of $\mathcal{S}_h(y)$ leads to $q = 6$ in the formula (8.17) for the starting procedure (the detailed proof is given in [122]);
- Equation (8.43) is satisfied with $\gamma = 8$ (detailed computations are again given in [122]).

Proposition 8.1 *If the method regarding this example is applied to a Hamiltonian system (1.22), then the Hamiltonian function is nearly preserved according to*

$$\mathcal{H}(y_n) - \mathcal{H}(y_0) = O(h^4) + O(th^8) + O\big(h^8 \exp(h^2 Lt)\big),$$

as long as $t = nh = O(h^{-2})$.

Proof The first two error terms follow directly from Theorem 8.19. From Theorem 8.17 we have that the parasitic solution components satisfy $z_{jj}(0) = O(h^6)$, so that $z_{jj}(t) = O\big(h^6 \exp(h^2 Lt)\big)$. To justify the factor h^8 in front of the exponential term we note that only the functions z_{1j} enter the formula for y_n. By symmetry of the method, we have a factor h^2 in the modified equation (8.31) for z_{1j}. This proves that $z_{1j}(t) = O\big(h^8 \exp(h^2 Lt)\big)$. □

Let us illustrate with numerical experiments that the bounds of Theorem 8.19 and, in particular, those for the parasitic solution components are sharp. In particular we aim to observe that, for multivalue methods for which the order q of the starting procedure is larger or equal than the order p of the method, the parasitic solution components can be neglected on time intervals of length $O(h^{-M})$. On such intervals the underlying one-step method completely describes the qualitative behavior of the method. In particular, if the problem is an integrable reversible system and if the underlying one-step method is symmetric (and reversible), then all action variables are preserved up to an error of size $O(h^p)$. Moreover, the global error increases at most linearly with time.

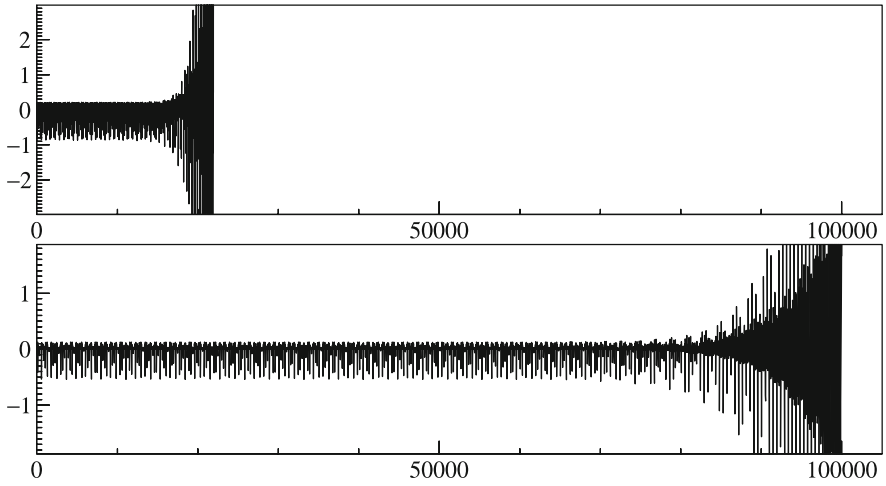


Fig. 8.15 Error in the Hamiltonian for the method in Example 8.8 applied to the mathematical pendulum (1.23), with initial values $q(0) = 3$, $p(0) = 0$. The employed values of h are $h = 0.25$ (top) and $h = 0.125$ (bottom)

Example 8.9 To prove that the estimate of Theorem 8.1 is sharp, we apply the method described in Example 8.8 to the mathematical pendulum (1.23), with initial values $q(0) = 3$, $p(0) = 0$. Figure 8.15 (see [122]) shows the error in the Hamiltonian as a function of time for the step sizes $h = 0.25$ and $h = 0.125$. The scales on the vertical axis differ by a factor 16, so that the $O(h^4)$ behavior of the error can be observed. As predicted by the estimate of Theorem 8.1 the error behaves like $O(h^4)$ on intervals of length $O(h^{-2})$, and then follows an exponential growth. We notice that halving the step size increases the interval of good energy preservation by a factor of 4. This confirms the factor h^2 in the exponential term. The constant L in the estimate, which depends on the problem and on the coefficients of the method, seems to be rather small.

8.8 Exercises

1. Prove that the symplectic Euler method (8.10) is symplectic. The proof requires similar arguments as those used to prove Theorem 8.5.

2. Prove that the implicit midpoint method applied to (1.22), i.e.,

$$y_{n+1} = y_n + hJ^{-1}\nabla\mathcal{H}\left(\frac{y_n + y_{n+1}}{2}\right).$$

   is a symplectic method.
3. Complete the proof of Theorem 8.9, by providing the requested algebraic manipulations.
4. With reference to Example 8.6, compute the modified differential equation associated to the implicit midpoint method (4.24).
5. As highlighted in [192], prove that symplectic Runge-Kutta methods preserve all invariants of the form

$$I(y) = y^{\intercal}Cy + d^{\intercal}y + c.$$

6. As remarked in the explanation of Fig. 8.4, a linear energy drift is visible for the explicit Euler method, that is a non-symplectic method. Give a proof of this fact, i.e.,

$$\mathcal{H}(y_n) = \mathcal{H}(y_0) + O(th^p),$$

   through similar arguments as those provided in the proof of Benettin-Giorgilli theorem (8.15).
7. By using Program 8.2, solve the non-separable Hamiltonian problem whose Hamiltonian is given by

$$\mathcal{H}(p,q) = \frac{p^2}{2(1+U'(q))} + U(q),$$

   being $U(q) = 0.1(q(q-2))^2 + 0.008q^3$, with initial values $p(0) = 0.49$ and $q(0) = 0$, describing the path of a particle of unit mass moving on a wire of shape $U(q)$ [15]. In the numerical solution, focus on the conservation of the Hamiltonian and comment the results.
8. By using Program 8.2, solve the separable Hamiltonian problem whose Hamiltonian is the following polynomial of degree 6 [164]:

$$\mathcal{H}(p,q) = \frac{p^3}{3} - \frac{p}{2} + \frac{q^6}{30} + \frac{q^4}{4} - \frac{q^3}{3} + \frac{1}{6},$$

   by choosing several initial values. In the numerical solution, focus on the conservation of the Hamiltonian and comment the results.

9. Can explicit Runge-Kutta methods be symmetric? Give a proof motivating your answer.
10. Prove that the underlying one-step method of a multivalue method cannot be symplectic. As aforementioned, proofs on non-symplecticity for multivalue method have been given in [71, 190, 250].

Chapter 9
Numerical Methods for Stochastic Differential Equations

They believe in chance because like themselves.

(James Joyce, Ulysses)

Stochastic differential equations, called "Itô Formula", are currently in wide use for describing phenomena of random fluctuations over time. When I first set forth stochastic differential equations, however, my paper did not attract attention. It was over ten years after my paper that other mathematicians began reading my "musical scores" and playing my "music" with their "instruments". By developing my "original musical scores" into more elaborate "music", these researchers have contributed greatly to developing Itô Formula.

(Kiyosi Itô [226])

This chapter is devoted to providing a bridge from the numerical discretization of deterministic differential equations to the case of stochastic differential equations, in order to both highlight basic accuracy and stability requirements and conservation issues along the numerical dynamics. The presentation in the direction of the numerics is rather self-contained, but previous knowledge of stochastic calculus is reasonably necessary. Useful comprehensive references (some of them also covering the numerical approximation of stochastic differential equations) are, for instance, [13, 17, 168, 173, 204, 213, 234, 237, 239, 259, 267–269, 280, 289, 309] and the other references therein. The gifted review paper [209] also provides a list of practical tools in the direction of an algorithmic introduction to the topic, together with a selection of Matlab programs from which we drew inspiration to design most of the codes reported in this section.

9.1 Discretization of the Brownian Motion

The basic theory of stochastic differential equations strongly relies on the notion of wiener process and its discretization. Let us first provide its definition.

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1_9

Definition 9.1 A scalar *standard Brownian motion* or *standard Wiener process* in the interval $[0, T]$ is a stochastic process $\{W(t),\ t \in [0, T]\}$ such that

1. $W(0) = 0$ with probability 1;
2. for any $0 \leq s < t \leq T$, the Wiener increment $W(t) - W(s)$ is a normally distributed random variable with zero mean and variance $t - s$. In symbols,

$$W(t) - W(s) \sim \sqrt{t - s}\, \mathcal{N}(0, 1),$$

   where $\mathcal{N}(0, 1)$ is a standard normal random variable;
3. for any $0 \leq s_1 < t_1 < s_2 < t_2 \leq T$, Wiener increments $W(t_1) - W(s_1)$ and $W(t_2) - W(s_2)$ are independent random variables.

As announced, we aim to provide a discretization of the Wiener process, i.e., a sampling of the random variable $W(t)$ evaluated in a discrete set of points in $[0, T]$. Therefore, we provide a partition of the interval $[0, T]$ in N subintervals of equal length

$$\delta t = \frac{T}{N}$$

and the corresponding set of sampled values is then given by

$$\{W_0 = W(\tau_0) = 0\} \cup \{W_j = W(\tau_j),\ j = 1, \ldots, N\},$$

with $\tau_j = j\delta t$, $j = 0, 1, \ldots, N$. As a consequence, according to Definition 9.1, the following recursion is established

$$\begin{aligned} W_0 &= 0, \\ W_j &= W_{j-1} + \Delta W_j, \quad j = 1, 2, \ldots, N, \end{aligned}$$

with $\Delta W_j \sim \mathcal{N}(0, \delta t)$. A graphic glance of Wiener increments referring to the introduced partition is available in Fig. 9.1.

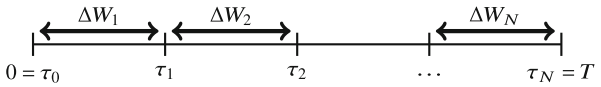


Fig. 9.1 A graphic glance of Wiener increments referring to the partition of the interval $[0, T]$ in N subintervals of equal length

The vector of Wiener increments $\Delta W = \left(\Delta W_j\right)_{j=1}^{N}$ is then given by

$$\Delta W = \sqrt{\delta t}\, v,$$

where v is a vector of N scalar normally distributed random variables. Then, the following proposition immediately holds true.

Proposition 9.1 *The vector $W = \left(W_j\right)_{j=0}^{N}$ of sampled values of $W(t)$ collects all cumulative summations of the vector ΔW, i.e.,*

$$W_0 = 0,$$

$$W_j = \sum_{k=1}^{j} \Delta W_k, \quad j = 1, 2, \ldots, N.$$

Proof The result can be proved by induction. For $j = 1$, we have $W_1 = W_0 + \Delta W_1 = \Delta W_1$. Supposing that

$$W_{j-1} = \sum_{k=1}^{j-1} \Delta W_k,$$

we have

$$W_j = W_{j-1} + \Delta W_j,$$

leading to the thesis. □

A Matlab code for the computation of a discretized Wiener process is given in Program 9.1. The pattern of a Wiener process computed as `W=wiener(1,1000)` is given in Fig. 9.2. As one can appreciate from the figure, the Wiener process is continuous but nowhere differentiable.

Program 9.1 (Discretized Wiener Process)

```
% Computation of a discretized Wiener process, according to
% Proposition 9.1

% Inputs:
% - T: maximum of the interval [0,T];
% - N: number of intervals.
```

(continued)

Program 9.1 (continued)

```
% Output:
% - dW: vector of Wiener increments;
% - W: discretized Wiener process.

function [dW,W]=wiener(T,N)
dt=T/N;
dW=sqrt(dt)*randn(1,N);
W=cumsum(dW);
W=[0 W];
plot(0:dt:T,W)
```

We now present a Matlab code for the computation of the expected value of a Wiener process over M samples. Since

$$W_j^i = W_{j-1}^i + \Delta W_j^i, \quad i = 1, 2, \ldots, M, \ j = 1, 2, \ldots, N,$$

with the Wiener increments ΔW_j^i, $i = 1, 2, \ldots, M$, $j = 1, 2, \ldots, N$, normally distributed random variables with mean zero and variance δt. Then, similarly as in Proposition 9.1,

$$\widetilde{W}_j^i = \sum_{k=1}^{j} \Delta W_k^i, \quad i = 1, 2, \ldots, M, \quad k = 1, 2, \ldots, N. \tag{9.1}$$

Let us now discuss how to provide the simultaneous computation of M Brownian paths, through the Matlab implementation proposed in the forthcoming Program 9.2.

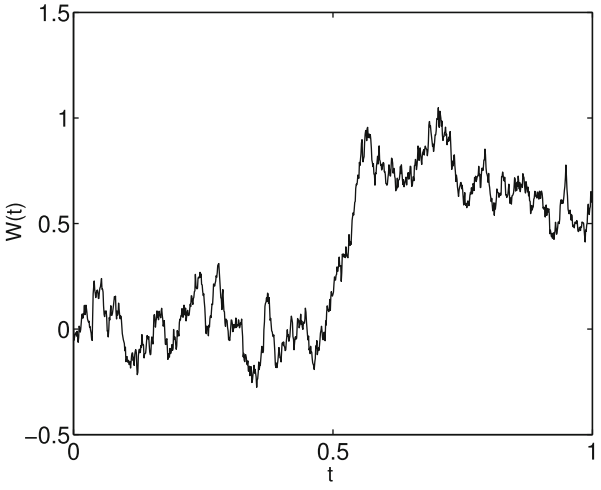


Fig. 9.2 A join-the-dots Wiener path obtained by Program 9.1 with $T = 1$ and $N = 1000$

In order to provide several samples of the Wiener process all-at-once, we aim to simulate the matrix

$$W = \begin{matrix} \tau_0 & \tau_1 & \tau_2 & \dots & \tau_N \end{matrix} \\ \begin{bmatrix} 0 & & & & \\ 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{bmatrix} \begin{matrix} \text{trajectory 1} \\ \text{trajectory 2} \\ \vdots \\ \text{trajectory } M \end{matrix}$$

where each row corresponds to a sampled Wiener trajectory and, correspondingly, each column stores the values of the simulated trajectory in a specific point of the discretization. Following Proposition 9.1, we immediately obtain that

$$W_j^i = \sum_{k=1}^{j} \Delta W_k^i, \quad i = 1, 2, \dots, M, \quad j = 1, 2, \dots, N,$$

i.e., the matrix W is the cumulative sum of the matrix $\Delta W \in \mathbb{R}^{M \times N}$ of (pathwise) Wiener increments, along its columns. An example of simultaneous generation of several Wiener paths, using Program 9.2, is given in Fig. 9.3

Program 9.2 (Discretized Wiener Process over Several Samples)

```
% Computation of a discretized Wiener process over several
% samples, according to Equation 9.1

% Inputs:
% - T: maximum of the interval [0,T];
% - M: number of realizations;
% - N: number of intervals.

% Output:
% - dW: Wiener increments over all the sampled paths;
% - W: matrix of M sampled Wiener paths.

function [dW,W]=manyWiener(T,M,N)
dt=T/N;
dW=sqrt(dt)*randn(M,N);
W=cumsum(dW,2);
W=[zeros(M,1) W];
plot(0:dt:T,W)
```

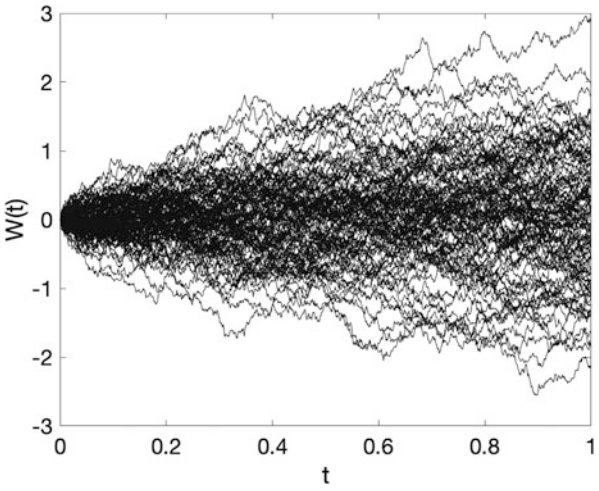


Fig. 9.3 A hundred Wiener paths computed through Program 9.2, with $N = 1000$ Wiener points

9.2 Itô and Stratonovich Integrals

We now aim to define the integral with respect to a Wiener process, in analogous way as in the case of deterministic Riemann integration. In other terms, for a given scalar function $h : [t_0, T] \to \mathbb{R}$, we aim to define the integral

$$I(h) = \int_{t_0}^{T} h(s)\mathrm{d}W(s). \tag{9.2}$$

The construction of this integral is now provided in two steps. We first define (9.2) when h is a step function

$$h(t) = h_j \in \mathbb{R}, \quad t \in [\tau_j, \tau_{j+1}), \quad j = 0, 1, \ldots, N-1,$$

with $t_0 = \tau_0 < \tau_1 < \ldots < \tau_N = T$. In this case, we define the *Itô integral* of the step function h by

$$I(h) = \sum_{j=0}^{N-1} h_j \left(W(\tau_{j+1}) - W(\tau_j)\right).$$

We now define the Itô integral of a generic continuous function $v : [t_0, T] \to \mathbb{R}$. To this purpose, we introduce the system of $n+1$ nodes

$$t_0 = t_0^{(n+1)} < t_1^{(n+1)} < \ldots < t_n^{(n+1)} = T, \quad n \geq 0$$

and give the following definition.

Definition 9.2 The *Itô integral* of a continuous function $v : [t_0, T] \to \mathbb{R}$ is defined as the limit of the integral of the raised step functions, i.e.,

$$I(v) = \lim_{n\to\infty} \sum_{j=0}^{n-1} v\left(t_j^{(n+1)}\right)\left(W(t_{j+1}^{(n+1)}) - W(t_j^{(n+1)})\right).$$

In other terms, given a discretization of the integration interval

$$t_0 < t_1 < \ldots < t_N = T,$$

the Itô integral of a continuous function can be approximated by the following *Itô quadrature formula*

$$\int_{t_0}^{T} v(t)\mathrm{d}W(t) \approx \sum_{j=0}^{N-1} v\left(t_j\right)\left(W(t_{j+1}) - W(t_j)\right), \tag{9.3}$$

involving a linear combination of values of the integrand function evaluated in the left-hand endpoint of each subinterval of the domain discretization.

Program 9.3 provides a Matlab coding of the Itô integral of a given function by applying the Itô quadrature formula (9.3).

Program 9.3 (Application of Itô Quadrature Formula)

```
% Computation of the Ito integral of a given function
% through Ito quadrature formula (9.3).

% Inputs:
% - T: maximum of the interval [0,T];
% - N: number of intervals of the discretization.

% Output:
% - itoIntegral: approximation of the Ito integral.

function itoIntegral=itoQuadrature(T,N)
[dW,W]=wiener(T,N);
t=linspace(0,T,N+1);
v=fun(t,W);
itoIntegral=v(1:end-1)*dW';
```

Let us briefly list some relevant properties of Itô integral; the reader can find more details in [17, 168, 204, 213, 234, 237, 239, 259, 267, 269, 280, 309] and references therein:

- any (even random) function $v(t)$ is Itô-integrable if it is non-anticipative, i.e., at time t it must be independent of the later values $\{W(s)\}_{s>t}$ of the Wiener process;
- the following *martingale property* holds true

$$\mathbb{E}\left[\int_0^T v(t)\mathrm{d}W(t)\right] = 0; \tag{9.4}$$

- the following property of *Itô isometry* allows a form of conversion from stochastic to standard Riemann integrals:

$$\mathbb{E}\left[\left(\int_0^T v(t)\mathrm{d}W(t)\right)^2\right] = \mathbb{E}\left[\int_0^T v(t)^2\mathrm{d}t\right]. \tag{9.5}$$

Let us now briefly present some biographical notes of Kiyosi Itô, based on the information reported in the MacTutor History of Mathematics Archive (https://mathshistory.st-andrews.ac.uk/Biographies/Ito/) and [226].

A Portrait of Kiyosi Itô

Kiyosi Itô was born in 1915 in Japan and he is certainly acknowledged as a pioneer in the theory of stochastic differential equations. He graduated in Mathematics in 1938, at the Imperial University of Tokyo, where he discovered his genuine passion for probability theory and, to some extent, the proper direction to follow [226]: "*At that time, few mathematicians regarded probability theory as an authentic mathematical field, in the same strict sense that they regarded differential and integral calculus. With clear definition of real numbers formulated at the end of the 19th century, differential and integral calculus had developed into an authentic mathematical system. When I was a student, there were few researchers in probability; among the few were Kolmogorov of Russia, and Paul Levy of France*".

After graduating, he worked in the Statistical Bureau of the Japanese Government until 1943. These were crucial years for his scientific contributions, since he had the opportunity to study in depth the pioneering papers of Kolmogorov and Levy released at that time and provide this first remarkable results on stochastic integration [225] in 1942, almost 20 years after the contributions of Wiener on probability measures.

In 1943 he got a position as Assistant Professor in Nagoya Imperial University. At that time, notwithstanding with the difficult times in Japan for

(continued)

World War II, he was highly prolific in his scientific work: volume 20 of the Proceedings of the Imperial Academy of Tokyo contains six papers authored by him.

In 1945 Itô was awarded his doctorate and appointed as Professor at Kyoto University in 1952. In 1954–1956 he visited the Institute for Advanced Study at Princeton University, leading to his book on stochastic processes published in 1957. He has also held professor positions at Aarhus University from 1966 to 1969 and Cornell University from 1969 to 1975, still remaining in Kyoto until his retirement in 1979.

Several prized were awarded to Ito: in 1978, he achieved the Asahi Prize, the Imperial Prize and also the Japan Academy Prize; in 1985 he received the Fujiwara Prize and in 1998 the Kyoto Prize in Basic Sciences from the Inamori Foundation; he was elected to the National Academy of Science of the United States and to the Académie des Sciences of France. He received the Wolf Prize from Israel and honorary doctorates from the universities of Warwick, England and ETH, Zürich, Switzerland. He won the IMU Gauss prize in 2006. He died in Kyoto in 2008.

An alternative definition of integral with respect to a Brownian motion is given by the so-called *Stratonovich integral*, defined as the limit of the following quadrature formula

$$\int_{t_0}^{T} v(t) \circ \mathrm{d}W(t) \approx \sum_{j=0}^{N-1} v\left(\frac{t_j + t_{j+1}}{2}\right)\left(W(t_{j+1}) - W(t_j)\right) \tag{9.6}$$

and is usually denoted by the symbol $\circ$ close to $\mathrm{d}W(t)$ in the integral. Unlike Itô case, Stratonovich quadrature formula (9.6) involves evaluations of the integrand function in the midpoint of each subinterval of the domain discretization.

A Matlab coding for the computation of the Stratonovich integral is object of Exercise 2 at the end of this chapter.

Let us now briefly provide some historical notes regarding Ruslan Leont'evich Stratonovich, based on [42].

A Portrait of Ruslan Leont'evich Stratonovich

Ruslan Leont'evich Stratonovich was born in Moscow in 1930. In 1947, after passing his school examinations with a gold medal, he started his studies at Moscow State University, in the Faculty of Physics, where he graduated in 1953. He first studied some problems of oscillation physics

(continued)

with P.I. Kuznetsov and then came into contact with the Kolmogorov. In 1956 he received his doctorate. His doctoral dissertation, establishing a theory of conditional Markov processes, was published as a monograph in 1966, with the title *Uslovnye Markovskie Protsessy i ikh Primenenie v Teorii Optimalnogo Upravleniya* (Conditional Markov Processes and Their Application to the Theory of Optimal Control). In the US edition, released in 1988, R. Bellman wrote in his foreword that *"Stratonovich's book represents a major step forward in the current endeavor to create unified mathematical theories with wide ranging applications in both mathematics itself and in science"*.

His candidate's dissertation was the core of his first monograph *Izbrannye Voprosy Teorii Fluktuatsiv Radiotekhnike* (Selected Topics of the Theory of Fluctuations in Telecommunications), published in the Soviet Union in 1961 and, few years later, in the United States under the title *Topics in the Theory of Random Noise* (in two volumes). In this monograph he developed what he defined a symmetrization form of integral and differential expressions for Markov processes and the stochastic calculus based on it, nowadays well known as Stratonovich calculus. In 1969 he became professor of physics at the Moscow State University.

Stratonovich also contributed in many more topics such as information theory, the theory of optimal statistical decisions and theory of optimal control, kinetic theory, quantum theory, statistical physics. Last topic was covered in his monograph, *Nelinenaya Neravnovesnaya Termodinamika* (Nonlinear Non-equilibrium Thermodynamics), first published in 1985 and then revised and enlarged in 1992 and 1994, as part of the Springer Series in Synergetics (vols. 57 and 63).

He was awarded with the Lomonosov Prize of Moscow University in 1984, a USSR State Prize in 1988, and a Russian Federation State Prize in 1996. He died in 1997.

We read in [42]: "*He loved and was well versed in Russian poetry, both classical and modern, wrote lyrics, was a connoisseur of painting, and could read fiction in four foreign languages. He was a true sports enthusiast: he went in for figure skating, tennis, and cycling. Sometimes he even went to work on his bicycle. Being a scientist of world renown, he remained a true friend of his disciples with no effort on his part, thus setting an example of sincerity and simplicity in personal relations*".

Itô and Stratonovich calculus, although defining the same object (i.e., the integral with respect to a Wiener process), are characterized by different properties and are both used in mathematical modeling. For instance, Stratonovich calculus does not obey to the martingale property as in the Itô case. Anyway, as proved for instance in

[239], Itô and Stratonovich integrals are related each other by the formula

$$\int_0^T f(W(t)) \circ \mathrm{d}W(t) = \frac{1}{2}\int_0^T \frac{\partial f}{\partial W}(W(t))\mathrm{d}t + \int_0^T f(W(t))\,\mathrm{d}W(t),$$

where f is any function of $W(t)$ of class C^1. In the remainder, unless differently specified, we will always refer to stochastic integrals as Itô integrals. We also highlight that Itô and Stratonovich calculus obey to different chain rules. This will be later clarified in the context of SDEs.

An example of computation of Itô and Stratonovich integrals is given in the following example, see [209].

Example 9.1 Let us apply the quadrature formulae (9.3) and (9.6) to compute Itô and Stratonovich integrals of the function $W(t)$. First of all, let us compute the exact value of both integrals by applying their definitions. As regards the Itô integral,

$$\begin{aligned}
\int_0^T W(t)\mathrm{d}W(t) &= \lim_{N\to\infty}\sum_{j=0}^{N-1} W(t_j)\left(W(t_{j+1}) - W(t_j)\right) \\
&= \frac{1}{2}\lim_{N\to\infty}\sum_{j=0}^{N-1}\left(W(t_{j+1})^2 - W(t_j)^2 - \left(W(t_{j+1}) - W(t_j)\right)^2\right) \\
&= \frac{1}{2}\lim_{N\to\infty}\left(\sum_{j=0}^{N-1}\left(W(t_{j+1})^2 - W(t_j)^2\right)\right. \\
&\qquad\left. - \sum_{j=0}^{N-1}\left(W(t_{j+1}) - W(t_j)\right)^2\right) \\
&= \frac{1}{2}W(T)^2 - \frac{1}{2}\lim_{N\to\infty}\sum_{j=0}^{N-1}\left(W(t_{j+1}) - W(t_j)\right)^2.
\end{aligned}$$

Let us now compute the expected value of the second summand of the last line, given by

$$\begin{aligned}
\mathbb{E}\left[\sum_{j=0}^{N-1}\left(W(t_{j+1}) - W(t_j)\right)^2\right] &= \sum_{j=0}^{N-1}\mathbb{E}\left[\left(W(t_{j+1}) - W(t_j)\right)^2\right] \\
&= \sum_{j=0}^{N-1}\left(t_{j+1} - t_j\right) = T.
\end{aligned}$$

(continued)

Example 9.1 (continued)
Moreover,

$$\mathbb{E}\left[\sum_{j=0}^{N-1}\left(W(t_{j+1})-W(t_j)\right)^2\right]=\sum_{j=0}^{N-1}\mathbb{E}\left[\left(W(t_{j+1})-W(t_j)\right)^2\right]$$
$$=\sum_{j=0}^{N-1}\left(t_{j+1}-t_j\right)=T$$

and

$$\lim_{N\to\infty}\mathrm{Var}\left[\sum_{j=0}^{N-1}\left(W(t_{j+1})-W(t_j)\right)^2\right]=0.$$

It follows that

$$\int_0^T W(t)\mathrm{d}W(t)=\frac{1}{2}W(T)^2-\frac{1}{2}T.$$

As regards the Stratonovich case, one can prove that

$$\int_0^T W(t)\circ\mathrm{d}W(t)=\frac{1}{2}W(T)^2.$$

Let us now compute the approximations arising from the quadrature formulae (9.3) and (9.6), by applying Programs 9.3 and the following lines of Matlab coding for the Stratonovich quadrature:

```
N=input('Number of Wiener points: ');
T=input('Maximum of the integration interval: ');
dt=T/N;
dW=sqrt(dt)*randn(1,N);
W=cumsum(dW);
ave = 0.5*([0,W(1:end-1)] + W(1:end));
Wave = ave + 0.5*sqrt(dt)*randn(1,N);
straIntegral=Wave*dW';
```

The results, reported in Table 9.1, provide a measure of the pathwise gap between the above computed exact values of the integrals and their approximations, for $T = 1$ and for several values of N. Clearly, as N changes at each run of the code needed to fill Table 9.1 in, new Wiener increments have

(continued)

Table 9.1 Example 9.1: pathwise errors associated to the approximation of Itô and Stratonovich integrals of $W(t)$ in [0,1], for several numbers of Wiener points, using quadrature formulae (9.3) and (9.6), respectively

N	Error in Itô integral	Error in Stratonovich integral
100	$1.72 \cdot 10^{-2}$	$5.45 \cdot 10^{-2}$
500	$2.68 \cdot 10^{-2}$	$1.23 \cdot 10^{-2}$
1000	$1.95 \cdot 10^{-2}$	$9.30 \cdot 10^{-3}$
10,000	$1.44 \cdot 10^{-2}$	$1.09 \cdot 10^{-2}$

Example 9.1 (continued)
to be randomly generated and, as a consequence, new paths of the Wiener process are computed for each value of N. As a consequence, one should not expect decreasing values of the errors for increasing values of N.

9.3 Stochastic Differential Equations

The development of the presentation that took place in the previous chapters has totally been devoted to understanding how to accurately approximate the solutions of initial value problems based on deterministic ODEs (1.1). For Hadamard well-posed problems, the dynamics is given with the initial value, in a purely deterministic way.

If the dynamics is governed by both deterministic and random forcing terms, Equation (1.1) is no longer enough to fully describe the underlying phenomenon and the corresponding model should incorporate the source of randomness in itself. To this purpose, *stochastic differential equations* (SDEs)

$$\begin{aligned} \mathrm{d}X(t) &= f(X(t))\mathrm{d}t + g(X(t))\,\mathrm{d}W(t), \quad t \geq 0, \\ X(0) &= X_0, \end{aligned} \tag{9.7}$$

are characterized by a right-hand side depending on two terms:

- the function $f : \mathbb{R}^d \to \mathbb{R}^d$, well-known as *drift* of the problem, that is the coefficient of its deterministic part;
- the function $g : \mathbb{R}^d \to \mathbb{R}^{d\times m}$, denoted in the literature as *diffusion* of the problem, that is the coefficient of its stochastic part.

The term $W(t)$ in (9.7) is a m-dimensional standard Wiener process and, due to its nowhere differentiability (with probability 1), the representation given in Eq. (9.7) is only a shorthand notation for its integral counterpart

$$X(t) = X(0) + \int_0^t f(X(s))\mathrm{d}s + \int_0^t g(X(s))\mathrm{d}W(s). \tag{9.8}$$

If the stochastic integral in the right-hand side of (9.8) is the Itô integral, then the corresponding equation is denoted as *Itô stochastic differential equation*; if it is the Stratonovich integral, Eq. (9.8) is a *Stratonovich stochastic differential equation*. Unless differently specified, our presentation is focused on Itô SDEs. We also observe that Itô SDEs admit an equivalent Stratonovich formulation

$$\mathrm{d}X(t) = \left(f(X(t)) - \frac{1}{2}g(X(t))g'(X(t))\right)\mathrm{d}t + g(X(t)) \circ \mathrm{d}W(t).$$

In other terms, we can pass from Itô to Stratonovich SDEs (and vice versa), obtaining formulations equivalent in terms of solution. However, the geometry of the problem may not be preserved by this transformation, as we discuss in Sect. 9.7.4: for instance Itô perturbation of Hamiltonian problems does not preserve the Hamiltonian function as it happens, on the contrary, to Stratonovich Hamiltonian problems.

If the function g in (9.8) is constant, the corresponding problem is denoted as SDE with *additive noise*; if g is solution dependent, (9.8) is a SDE with *multiplicative noise*.

Mathematical modeling is extremely rich in stochastic models, which cover a wide selection of fields insisting in scientific knowledge (see, for instance, [101, 102, 169, 176, 204, 210, 213, 219, 237, 240, 259, 269, 280, 290]): we cite, but only as a non-exhaustive list of examples, models in financial, biological, medical, physical and economic fields, as well as in population dynamics, in the description of opinion formation, in network analysis.

Example 9.2 (Geometric Brownian Motion) Let consider the case of linear drift and diffusion in (9.7), i.e.,

$$\begin{aligned} \mathrm{d}X(t) &= \mu X(t)\mathrm{d}t + \sigma X(t)\,\mathrm{d}W(t), \quad \mu, \sigma \in \mathbb{R}, \\ X(0) &= X_0. \end{aligned} \tag{9.9}$$

well known in the literature as the equation of *geometric Brownian motion*. It is a scalar SDE with linear multiplicative noise, appearing as a stochastic perturbation of Dahlquist test problem (6.1). As we will see in Sect. 9.6, this

(continued)

Example 9.2 (continued)
is the test equation for the analysis of linear stability in the numerics for SDEs. However, this equation is very relevant in the context of financial mathematics, since it models the evolution of a stock price in the Black-Scholes theory for financial option evaluation.

The coefficient σ of the part modelling random fluctuations is known in the literature as *volatility*. If the volatility is equal to 0 and the initial value X_0 is deterministic, then the corresponding model $X'(t) = \mu X(t)$ is purely deterministic and describes a non-risky deposit in a bank with interest rate μ; if $\sigma \neq 0$, the exact solution is given by

$$X(t) = X_0 \mathrm{e}^{(\mu - \frac{1}{2}\sigma^2)t + \sigma W(t)}$$

and one can easily prove that its expected value is

$$\mathbb{E}[X(t)] = \mathbb{E}[X_0]\mathrm{e}^{\mu t}.$$

and μ is the expected growth rate in the stock price model. Let us also observe that μ is the rate of exponential growth in the exact solution $X(t) = X_0 \mathrm{e}^{\mu t}$ of the purely deterministic model (i.e., Eq. (9.9) with $\sigma = 0$).

Figure 9.4 shows ten sampled trajectories of (9.9), for selected values of σ and the reference solution of the underlying deterministic problem, for $\sigma = 0$. We can appreciate from the figure that larger values of σ provide more jagged paths tending to spread further from the mean. Clearly, larger values of σ make the stochasticity more dominant in the model. The consequence of this issue, also under the numerical point of view, will be clarified in the following sections.

An alternative model is given by the so-called *mean-reverting square root process*, described by

$$\begin{aligned} &\mathrm{d}X(t) = \lambda(\mu - X(t))\mathrm{d}t + \sigma\sqrt{X(t)}\mathrm{d}W(t), \quad \lambda, \mu, \sigma > 0, \\ &X(0) = X_0. \end{aligned}$$

This model is used in mathematical finance as an alternative to geometric Brownian motion, since the presence of the square root dampens the influence of the noise for large values of $X(t)$, that appears to be pretty more realistic. For this model, the dynamics over long time windows can be well clarified by the long-term expectation

$$\lim_{t \to \infty} \mathbb{E}[X(t)] = \mu,$$

(continued)

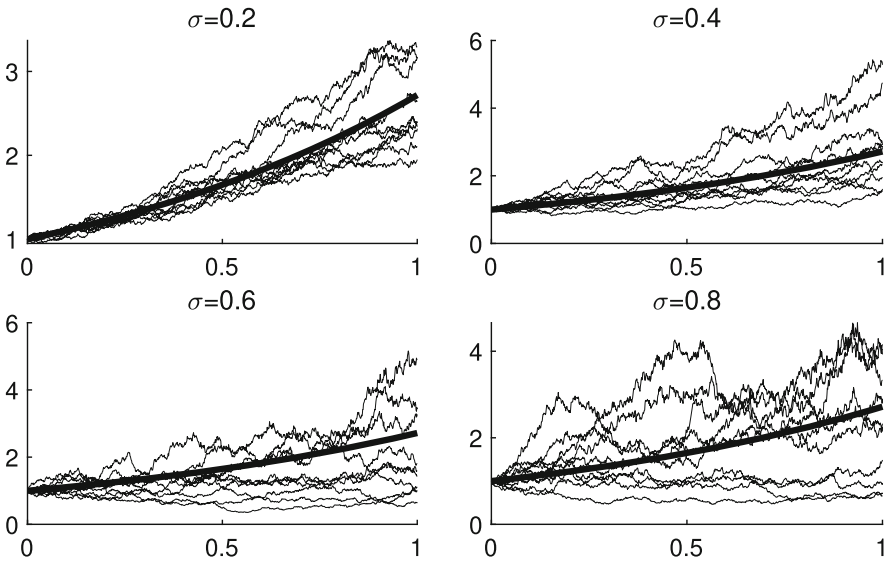


Fig. 9.4 Example 9.2: solutions of (9.9), for $X_0 = 1$, $\mu = 1$ and selected values of σ. Each subplot displays ten sampled trajectories, together with the solution of the underlying deterministic equation $X'(t) = X(t)$, with $X_0 = 1$, drawn with a thicker line

Example 9.2 (continued)
equal to the parameter μ, and the long-term variance

$$\lim_{t\to\infty} \mathbb{E}[X(t)^2] = \mu^2 + \frac{\sigma^2\mu}{2\lambda},$$

where we can appreciate that λ is the rate of convergence for the mean and the noise coefficient σ affects the variance.

Another important issue is the analysis of non-negativity of the solution [310], since it is possible to prove that, if $\mathbb{P}(X_0 \geq 0) = 1$, then $\mathbb{P}(X(t) \geq 0) = 1$, for any t and the solution attains the value zero if and only if $\sigma^2 > 2\lambda\mu$.

Example 9.3 (Stochastic Ginzburg-Landau Equation) Let us consider the following stochastic Ginzburg-Landau model [174, 222]

$$\mathrm{d}X(t) = \left(\beta X(t) - \gamma X(t)^3\right)\mathrm{d}t + \delta X(t)\mathrm{d}W(t), \quad \beta, \gamma, \delta \in \mathbb{R}, \tag{9.10}$$

(continued)

Example 9.3 (continued)
of phase transition in superconductivity theory. It is possible to prove (see [239]) that, under suitable regularity assumptions, two solutions $X(t)$ and $Y(t)$ of (9.10), computed in correspondence of two distinct initial values X_0 and Y_0, satisfy the following inequality

$$\mathbb{E}\left[\|X(t) - Y(t)\|^2\right] \leq \mathbb{E}\left[\|X_0 - Y_0\|^2\right] e^{\alpha t}, \quad \alpha < 0,$$

where α is a constant value, depending on the drift and the diffusion of (9.10), whose role will be clarified in Sect. 9.7.1. Clearly, if $\alpha < 0$, the gap between $X(t)$ and $Y(t)$ gets damped in time: in other terms, in analogy with the corresponding deterministic concept described in Chap. 1, this situation leads to *mean-square dissipativity* of the problem and *mean-square contractivity* in its solutions [38, 117, 118]. These aspects will be treated in detail in Sect. 9.7.1, as a good prototype of test problem for the analysis of nonlinear stability issues.

Example 9.4 (Itô-Hamiltonian Problems) Starting from a general Hamiltonian problem (1.22), we consider the following stochastic Hamiltonian system of Itô type [85]

$$\begin{cases} \mathrm{d}q(t) = \nabla_p H(q(t), p(t))\mathrm{d}t, \\ \mathrm{d}p(t) = -\nabla_q H(q(t), p(t))\mathrm{d}t + \Sigma \mathrm{d}W(t), \end{cases} \tag{9.11}$$

where $\Sigma \in \mathbb{R}^{d\times m}$, whose generic element is denoted by σ_{ij}, for $i = 1, 2, \ldots, d$ and $j = 1, 2, \ldots, m$. Moreover, we assume that the initial datum (q_0, p_0) of the system (9.11) provides an initial Hamiltonian of finite expectation, i.e., $\mathbb{E}[H(q_0, p_0)] < \infty$. We note that if the matrix Σ is the zero matrix, the stochastic Hamiltonian system (9.11) recasts the deterministic Hamiltonian system whose Hamiltonian is conserved along its exact flow.

Stochastic Hamiltonian problems are able to conjugate the canonical character of evolution equations (the Hamiltonian description of motion) with the stochastic effects visible, for instance, in the statistical independence of the future from the past and the irreversibility of the time arrow, making the resulting equations of motion more realistic models [21, 247, 333].

It is possible to prove (see [47, 48, 85, 121] and references therein) that, along the dynamics of Itô-Hamiltonian problems (9.13), the Hamiltonian is not preserved, nor pathwise or in expectation. Indeed, the following

(continued)

Example 9.4 (continued)
expression of the expectation of the Hamiltonian at time $t \in [0, T]$ is established [48]

$$\mathbb{E}[H(q(t), p(t))] = \mathbb{E}[H(q_0, p_0)] + \frac{1}{2}\sum_{i=1}^{m} \bar{\sigma}_i^2 \int_0^t \mathbb{E}\left[\nabla_{pp}^{ii} H(q(s), p(s))\right] \mathrm{d}s,$$

where $\bar{\sigma}_i$ denotes the diagonal element on the i-th row of the matrix $\Sigma^\intercal \Sigma$, $i = 1, \ldots, d$, and by $\nabla_{pp}^{ii} H$ the element in position (i, i) of the Hessian matrix associated to the function H, computed with respect to p.

In the case of separable Hamiltonian functions of type

$$H(q, p) = \frac{1}{2}\sum_{i=1}^{d} p_i^2 + V(q), \tag{9.12}$$

depending on a suitable smooth potential $V : \mathbb{R}^d \to \mathbb{R}$, the corresponding stochastic Hamiltonian system of Itô type (9.11) reads

$$\begin{cases} \mathrm{d}q(t) = p(t)\mathrm{d}t, \\ \mathrm{d}p(t) = -\nabla_q V(q(t))\mathrm{d}t + \Sigma \mathrm{d}W(t). \end{cases} \tag{9.13}$$

Correspondingly, the expected Hamiltonian assumes a more compact form [48, 85]

$$\mathbb{E}[H(q(t), p(t))] = \mathbb{E}[H(q_0, p_0)] + \frac{1}{2}\mathrm{Tr}\left(\Sigma^\intercal \Sigma\right) t, \tag{9.14}$$

known in the literature as *trace equation*. This formula reveals that, for the Hamiltonian system (9.13), the expectation of the Hamiltonian function (9.12) grows linearly in time and the growth rate depends on the trace of the matrix $\Sigma^\intercal \Sigma$. The analysis of the conservative character of numerical methods applied to stochastic Hamiltonian problems will be treated in Sect. 9.7.4.

We finally report a relevant result, concerning the existence and uniqueness of solutions to (9.7). A complete proof is given, for instance, in [239].

Theorem 9.1 *Suppose that, for a given SDE (9.7), the following conditions hold true:*

- *the drift f and the diffusion g are L^2-measurable in $\mathbb{R}^d$;*
- *f and g are Lipschitz functions, i.e., there exists a positive constant K such that*

$$\max\{\|f(x) - f(y)\|, \|g(x) - g(y)\|\} \leq K\|x - y\|,$$

for any $x, y \in \mathbb{R}^d$;
- *the following linear growth bound is satisfied by f and g: there exists a positive constant Λ such that*

$$\max\left\{\|f(x)\|^2, \|g(x)\|^2\right\} \leq \Lambda^2(1 + \|x\|^2), \tag{9.15}$$

for any $x \in \mathbb{R}^d$;
- *the initial value X_0 has bounded $\mathbb{E}\left[\|X_0\|^2\right]$.*

Then, the SDE (9.7) has a pathwise unique solution $X(t)$ in $[0, +\infty)$, called Itô process, with

$$\sup_{t\geq 0} \mathbb{E}\left[\|X(t)\|^2\right] < \infty.$$

9.4 One-Step Methods

In developing numerical methods for ODEs (1.1), we have ascertained the effectiveness of some tools, such as Taylor expansions and numerical quadrature, useful to find the discretized counterpart of the continuous operator under investigation. It is now worth understanding which may be the stochastic counterpart of these tools, with particular reference to Taylor expansion, in order to provide an effective way to develop numerical methods.

As we have seen, for instance, for the development of the Butcher theory of order in Chap. 4, a basic tool is given by the chain rule. Indeed, the Taylor expansion of $y(t)$, solution of $y'(t) = f(y(t))$, given by

$$y(t + h) = y(t) + hy'(t) + \frac{h^2}{2}y''(t) + \ldots,$$

relies on the computation of

$$y'(t) = f(y(t)), \quad y''(t) = \frac{\mathrm{d}}{\mathrm{d}t} f(y(t)) = f'(y(t))y'(t) = f'(y(t))f(y(t))$$

and so on.

Let us present the chain rule of Itô calculus for the scalar case (its extension to the multi-dimensional case as well as its proof are discussed, for instance, in [239]). For a given Itô process $X(t)$ and a function $v : \mathbb{R} \to \mathbb{R}$, $v(X(t))$ is still an Itô process, satisfying the SDE

$$\begin{aligned} v(X(t)) = &\int_0^t \left(v'(X(s)) f(X(s)) + \frac{1}{2} v''(X(s)) g^2(X(s)) \right) \mathrm{d}s \\ &+ \int_0^t v'(X(s)) g(X(s)) \, \mathrm{d}W(s), \quad t \geq 0, \end{aligned} \tag{9.16}$$

giving the chain rule of Itô calculus, better known in the literature as *Itô formula*.

Let us observe that Stratonovich calculus obeys to the classical chain rule of real calculus. The fact that Itô and Stratonovich calculi have different chain rules determines a number of different issues, especially in the conservative character of the corresponding SDEs, as described in Sect. 9.7.

9.4.1 Euler-Maruyama and Milstein Methods

The full discretization of a stochastic differential equations (9.7) requires performing several steps: first of all, the discretization of the Brownian motion, as we have discussed in the previous sections; then, as usual also in the case of ODEs, the discretization of the domain. To this purpose, supposing that the equation is studied in the closed interval $[0, T]$, let us partition it in L subintervals of equal length

$$\Delta t = \frac{T}{L},$$

intercepting the corresponding set of grid points

$$\mathcal{I}_{\Delta t} = \left\{ t_j = j\Delta t, \ j = 0, 1, \ldots, L \right\}. \tag{9.17}$$

Generally, such points are chosen as a subset of Wiener points; in other terms,

$$\Delta t = R\delta t, \quad R \in \mathbb{N}.$$

Let us now understand how to advance from a given point t_n to the subsequent point t_{n+1}. To this purpose, we first consider the integral formulation (9.8) for $t \geq t_n$,

i.e.,

$$X(t) = X(t_n) + \int_{t_n}^{t} f(X(s))\mathrm{d}s + \int_{t_n}^{t} g(X(s))\mathrm{d}W(s).$$

Let us now apply Itô formula (9.16), for v equal to the drift function f, i.e.,

$$f(X(s)) = f(X(t_n)) + \int_{t_n}^{s} \left(f'(X(u))f(X(u)) + \frac{1}{2} f''(X(u))g^2(X(u)) \right) \mathrm{d}u + \int_{t_n}^{s} f'(X(u))g(X(u))\,\mathrm{d}W(u)$$

and the diffusion function g, i.e.,

$$g(X(s)) = g(X(t_n)) + \int_{t_n}^{s} \left(g'(X(u))f(X(u)) + \frac{1}{2} g''(X(u))g^2(X(u)) \right) \mathrm{d}u + \int_{t_n}^{s} g'(X(u))g(X(u))\,\mathrm{d}W(u).$$

Let us truncate both equations to the first order, i.e., approximate

$$f(X(s)) \approx f(X(t_n)), \quad g(X(s)) \approx g(X(t_n))$$

and replace these values in (9.8) obtaining

$$\begin{aligned} X(t) &\approx X(t_n) + \int_{t_n}^{t} f(X(t_n))\mathrm{d}s + \int_{t_n}^{t} g(X(t_n))\mathrm{d}W(s) \\ &\approx X(t_n) + f(X(t_n))(t - t_n) + g(X(t_n))(W(t) - W(t_n)). \end{aligned}$$

We finally evaluate last relation for $t = t_{n+1}$, leading to

$$X(t_{n+1}) \approx X(t_n) + f(X(t_n))\Delta t + g(X(t_n))\Delta W_{n+1},$$

having denoted $W_{n+1} = W(t_{n+1}) - W(t_n)$. Finally, defining $X_n :\approx X(t_n)$ yields

$$X_{n+1} = X_n + f(X_n)\Delta t + g(X_n)\Delta W_{n+1}, \tag{9.18}$$

that is the famous *Euler-Maruyama method* for the numerical solution of SDEs (9.8).

As the name itself suggests, this numerical method arises as stochastic perturbation of the explicit Euler scheme (2.19) for ODEs: indeed, for $g = 0$, we recover the deterministic Euler method. The denomination Euler-Maruyama method is then

the *summa* of the work of Euler (1707–1783) and Gisiro Maruyama (1916–1986), whose portrait is now briefly presented, based on [322, 334].

A Portrait of Gisiro Maruyama

Gisiro Maruyama was born in Japan in 1916 and graduated from Tohoku Imperial University in 1939. His first interest was Fourier analysis, certainly motivated by the active environment he found in Tohoku at that time and, indeed, his first paper written in 1939 was focused on that topic. At a certain point, impressed by the papers of Norbert Wiener, he developed a genuine interest in probability theory. Influenced by several papers of Slutsky, Wiener, Wold and Hopf, he studied stationary processes and wrote a seminal paper on the topic, appeared in 1949 (actually, in Japan had already appeared in 1947), after which he got the degree of Doctor of Science. Certainly, his previous interest in Fourier analysis had a deep influence in his following results on probability theory.

He was appointed as a research assistant in Kyushu University in 1941 and was promoted to a professor position in 1949. Later he served as a professor in several universities, namely in Ochanomizu University, Kyushu University (for the second time), Tokyo University of Education, the University of Tokyo, the University of Electro-Communications, and finally in Tokyo Denki University where he remained until his death, occurred in 1986.

A key role in his scientific production was certainly played by the paper of Kiyosi Itô on stochastic differential equations, published in 1942. As we read in [334], "*Maruyama immediately recognized the importance of this work and soon published a series of papers on stochastic differential equations and Markov processes*". The study of convergence properties of numerical discretizations to stochastic differential equations, published in 1955, is certainly one of the masterpieces of Maruyama, now everywhere acknowledged as Euler-Maruyama method.

Euler-Maruyama method (9.18) is a one-step method, with explicit structure, then very easy to use (for this reason, this scheme is very well known also outside the mathematical community). It arises from a direct application of Itô formula (9.16), truncated to the very first term. Repeated application of (9.16) to all occurrences of the drift and the diffusion of (9.8) gives the so-called *Itô-Taylor expansion* of the exact solution to (9.8) [239]. Then, Euler-Maruyama method arises as a first order truncation of the *Itô-Taylor expansion* of the exact solution to (9.8). Program 9.4 presents a Matlab implementation of this method.

Program 9.4 (Euler-Maruyama Method)

```
% Approximate solution of (9.8) via Euler-Maruyama method

% Inputs:
% - T: maximum of the interval [0,T];
% - X0: initial value;
% - N: number of intervals in Wiener discretization;
% - R: ratio of Euler-Maruyama and Wiener stepsizes.

% Output:
% - X: Euler-Maruyama approximate solution.

function X=eulerMaruyama(T,X0,N,R)
[dW,~]=wiener(T,N);
dt=T/N;
Dt=R*dt;
L=N/R;
X=zeros(1,L+1);
X(:,1)=X0;
for j=1:L
    Winc=sum(dW((j-1)*R+1:j*R));
    X(:,j+1)=X(:,j)+Dt*f(X(:,j))+g(X(:,j))*Winc;
end
plot(0:Dt:T,X)
```

We observe that, as visible in Program 9.4, adapting the discretization chosen for the computation of the solution to the original Wiener discretization is a crucial point. Let us focus on the computation of the Wiener increment related to a single Euler-Maruyama step from t_n to t_{n+1}, i.e.,

$$\begin{aligned} W(t_{n+1}) - W(t_n) &= W((n+1)\Delta t) - W(n\Delta t) \\ &= W((n+1)R\delta t) - W(nR\delta t) \\ &= \sum_{k=1}^{(n+1)R} \Delta W_k - \sum_{k=1}^{nR} \Delta W_k \\ &= \sum_{k=nR+1}^{(n+1)R} \Delta W_k, \end{aligned}$$

motivating the formula for `Winc` in Program 9.4. Let us now provide an example of Euler-Maruyama approximation to a given SDE, namely the equation of the geometric Brownian motion (9.9).

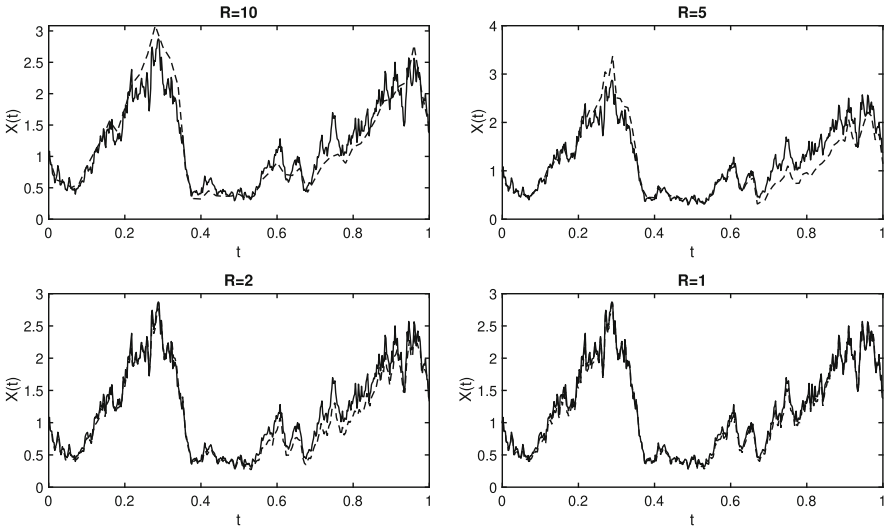


Fig. 9.5 Euler-Maruyama (dashed lines) vs exact solution (solid lines) of the geometric Brownian motion equation (9.9) in $[0, 1]$, for $\mu = 2$, $\sigma = 1$, with initial value X_0, $N = 500$ Wiener increments and various values of R

Example 9.5 Let us use Euler-Maruyama method (9.18) for the numerical solution of the equation of the geometric Brownian motion (9.9). Figure 9.5 shows the pattern of the sampled approximate trajectories in comparison with the exact solution, for various values of R. We can appreciate that, for a fixed Wiener path, decreasing the stepsize of the discretization (up to considering as grid points all Wiener points) makes the corresponding numerical solution closer to the exact one. It appears familiar to us: it looks like a form of convergence of Euler-Maruyama method. This issue will be analyzed in next section.

Let us conclude this part highlighting that, as for deterministic numerical methods, other formulae for the approximation of SDEs (9.8) may arise by incorporating further terms of Itô-Taylor expansion in the numerical method. This is the case of the so-called *Milstein method* that, for scalar problems, reads

$$X_{n+1} = X_n + f(X_n)\Delta t + g(X_n)\Delta W_{n+1} + \frac{1}{2}g'(X_n)g(X_n)(\Delta W_{n+1}^2 - \Delta t), \tag{9.19}$$

while, for systems of SDEs

$$\mathrm{d}X(t) = g^0(X(t))\mathrm{d}t + \sum_{k=1}^{m} g^k(X(t))\mathrm{d}W^k(t),$$

being $g^k : \mathbb{R}^d \to \mathbb{R}^d$, $k = 0, 1, \ldots, m$, the method is given by

$$X_{n+1} = X_n + g^0(X_n)\Delta t + \sum_{k=1}^{m} g^k(X_n)\Delta W_{n+1}^k + \sum_{j_1, j_2=1}^{m} L^{j_1} g^{j_2}(X_n) \int_{t_n}^{t_{n+1}} \int_{t_n}^{t} \mathrm{d}W^{j_1}(s)\mathrm{d}W^{j_2}(t)$$

with

$$L^k = \sum_{i=1}^{d} g^{k,i} \frac{\partial}{\partial x_i}, \quad k = 1, 2, \ldots, m.$$

Also Milstein method is a one-step explicit method and the presence of additional terms arising from the truncation of Itô-Taylor expansion makes the method more accurate than Euler-Maruyama method. This aspect will be analyzed in the next section.

9.4.2 Stochastic ϑ-Methods

A larger family of one-step methods for SDEs (9.8) is certainly given by the class of *stochastic ϑ-methods*, developed in order to especially improve the linear stability properties of Euler-Maruyama method, as we will discuss in Sect. 9.6. We distinguish two classes of stochastic ϑ-methods [37, 95, 97, 117, 119, 120, 127, 129, 208, 213, 215]: *stochastic ϑ-Maruyama methods* and *stochastic ϑ-Milstein methods*.

The family of stochastic ϑ-Maruyama methods is characterized by the same diffusion part as in Euler-Maruyama method (9.18) and arises from the following quadrature formula to approximate the drift part of the equation

$$\int_{t_n}^{t_{n+1}} f(X(t))\mathrm{d}t \approx \big((1-\vartheta) f(X(t_n)) + \vartheta f(X(t_{n+1}))\big)\Delta t,$$

with $\vartheta \in [0, 1]$. In other terms, the integral of the drift in $[t_n, t_{n+1}]$ is a convex combination of the value of $f(X(t_n))$ and $f(X(t_{n+1}))$; equivalently, the integrand is approximated by a linear interpolant. The corresponding numerical method is then given by

$$X_{n+1} = X_n + (1-\vartheta)\Delta t f(X_n) + \vartheta \Delta t f(X_{n+1}) + g(X_n)\Delta W_{n+1}. \tag{9.20}$$

This is a one-parameter family of one-step methods, depending on the parameter $\vartheta \in [0, 1]$. If $\vartheta = 0$, we recast the Euler-Maruyama method (9.18), that is the only explicit ϑ-Maruyama method. All methods (9.20) for $\vartheta \neq 0$ are implicit and it is worth highlighting two main cases:

- for $\vartheta = \frac{1}{2}$ we obtain the *stochastic trapezoidal method*

$$X_{n+1} = X_n + (f(X_n) + f(X_{n+1})) \frac{\Delta t}{2} + g(X_n)\Delta W_{n+1}; \tag{9.21}$$

- for $\vartheta = 1$ we obtain the *implicit Euler-Maruyama method*

$$X_{n+1} = X_n + f(X_{n+1})\Delta t + g(X_n)\Delta W_{n+1}. \tag{9.22}$$

An important matter that will be analyzed in Sect. 9.6 regards the optimal choice of the parameter ϑ in order to achieve good stability properties. In the implicit case, a nonlinear system for the computation of the updated solution is requested at each step. Under global Lipschitz conditions on the functions f and g, an application of the Banach fixed-point theorem ensures that a sufficient condition for the existence and uniqueness of the solution to such a nonlinear equation is given by $\sqrt{K}\Delta t < 1$ with probability 1, where K is the Lipschitz constant defined in Theorem 9.1. Weaker conditions are also admissible, e.g., a one-sided Lipschitz condition on the drift. If μ is the one-sided Lipschitz constant of f, then a sufficient condition for the existence and the uniqueness of the solution to (9.20) is given by $\mu\Delta t < 1$, see [208, 213, 259].

We also observe that, if the diffusion g in (9.20) is identically null, then stochastic ϑ-Maruyama methods perfectly overlap the family of deterministic ϑ-methods

$$X_{n+1} = X_n + \big((1-\vartheta) f(X_n) + \vartheta f(X_{n+1})\big)\Delta t.$$

We conclude this part by presenting the family of stochastic ϑ-Milstein methods developed in [37] that, for the scalar case, reads

$$\begin{aligned} X_{n+1} = X_n &+ (1-\vartheta)\Delta t f(X_n) + \vartheta \Delta t f(X_{n+1}) + g(X_n)\Delta W_{n+1} \\ &+ \frac{1}{2} g'(X_n) g(X_n)(\Delta W_{n+1}^2 - \Delta t), \end{aligned} \tag{9.23}$$

with $\vartheta \in [0, 1]$, sharing the diffusion part with Milstein method (9.19).

A Matlab coding for the ϑ-Maruyama method (9.20) is provided in Program 9.5. As in previous implementations of implicit methods, this coding also uses the built-in function `fsolve` to handle the implicitness of (9.20).

Program 9.5 (ϑ-Maruyama Method)

```
% Approximate solution of (9.8) via theta-Maruyama method

% Inputs:
% - T: maximum of the interval [0,T];
% - X0: initial value;
% - N: number of intervals in Wiener discretization;
% - R: ratio of Euler-Maruyama and Wiener stepsizes;
% - th: value of the parameter theta.

% Output:
% - X: theta-Maruyama approximate solution.

function X=thetaMaruyama(T,X0,N,R,th)
[dW,~]=wiener(T,N);
dt=T/N;
Dt=R*dt;
L=N/R;
X=zeros(1,L+1);
X(:,1)=X0;
options=optimset('Display','off','TolFun',eps,'TolX',eps);
for j=1:L
    Winc=sum(dW((j-1)*R+1:j*R));
    X(:,j+1)=fsolve(@(Y) Y-X(:,j)-(1-th)*Dt*f(X(,:j))...
            -th*Dt*f(Y)-g(X(:,j))*Winc,options);
end
plot(0:Dt:T,X)
```

9.4.3 *Stochastic Perturbation of Runge-Kutta Methods*

Stochastic differential equations (9.7) can be interpreted, to some extent, as a perturbation of ordinary differential equations (1.1), via a random forcing term governed by one or more Wiener processes. As a consequence, a natural question may be the following: can we obtain stochastic numerical methods as proper perturbations of deterministic ones?

A class of *stochastic Runge-Kutta methods* (SRK methods) has been obtained via proper perturbations of deterministic RK methods (4.8). This class has been introduced in [168, 303] and further analyzed (and in some cases enlarged) in [39, 46–48, 52, 118, 299–301] and reference therein. The formulation we use relies on the formalism in [168, 303], that represents SRK methods as follows:

$$X_{n+1} = X_n + \Delta t \sum_{i=1}^{s} b_i f(\widehat{X}_i) + \Delta W_{n+1} \sum_{i=1}^{s} q_i g(\widehat{X}_i), \tag{9.24}$$

where the internal stages $\widehat{X}_i$, approximating the value of $X(t_n + c_i \Delta t)$ for any index $i = 1, 2, \ldots, s$, are given by

$$\widehat{X}_i = X_n + \Delta t \sum_{j=1}^{s} a_{ij} f(\widehat{X}_j) + \Delta W_{n+1} \sum_{j=1}^{s} \gamma_{ij} g(\widehat{X}_j). \tag{9.25}$$

As already seen in the deterministic case, a more compact effective representation of SRK methods (9.24)–(9.25) is obtained by using a Butcher tableau that, in this case, reads

$$\begin{array}{c|c|c} c & A & \Gamma \\ \hline & b^{\intercal} & q^{\intercal} \end{array} = \begin{array}{c|ccc|ccc} c_1 & a_{11} & \ldots & a_{1s} & \gamma_{11} & \ldots & \gamma_{1s} \\ c_2 & a_{21} & \ldots & a_{2s} & \gamma_{21} & \ldots & \gamma_{2s} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \ldots & a_{ss} & \gamma_{s1} & \ldots & \gamma_{ss} \\ \hline & b_1 & \ldots & b_s & q_1 & \ldots & q_s \end{array},$$

with $A, \Gamma \in \mathbb{R}^{s \times s}$ and vector of weights $b, q \in \mathbb{R}^s$ and vector of the abscissae $c \in \mathbb{R}^s$.

If Γ is the zero matrix and q the null vector, then SRK methods (9.24)–(9.25) recover the analogous family of deterministic Runge-Kutta methods (4.8). Also in the stochastic case, the computational cost of the method is dictated by the structure of the matrices in (9.25): in particular, if A and Γ are strictly lower triangular, the corresponding method is explicit.

We also highlight that a two-step extension of SRK methods as stochastic perturbation of (5.15) has been given in [128], where the accuracy and stability analysis of this novel class of methods has also been given.

9.5 Accuracy Analysis

The transit from deterministic to stochastic numerics for evolutive problems does not alter the importance of analyzing proper accuracy and stability features of the developed scheme. In particular, in this section we provide concepts of convergence for numerical methods approximating SDEs (9.7) and investigate these convergence properties for the aforementioned numerical schemes.

A numerical method for SDEs (9.7) provides a sequence of random variables $\{X_n\}_{n=0}^{L}$, whose general term X_n approximates the exact solution $X(t_n)$ of (9.7). As in the deterministic setting, we aim to let the random variable X_n approach the exact value $X(t_n)$, for $\Delta t \to 0$ or, equivalently, the gap $\|X_n - X(t_n)\|$ should be infinitesimal with Δt.

The way a notion of convergence can be defined in the stochastic setting cannot be purely and simply inherited from the deterministic scenario. This is because the error $\|X_n - X(t_n)\|$ itself is a random variable, hence randomly fluctuating. A natural measure of the error may then be represented by the use of the expectation operator (see, for instance, [209, 213, 239] the references therein), as follows.

Definition 9.3 With reference to the discretization (9.17), a numerical method for SDEs (9.7) providing the numerical solution $\{X_n\}_{n=0}^{L}$ is *strongly convergent* if

$$\lim_{\Delta t \to 0} \sup_{t_n \in \mathcal{I}_{\Delta t}} \mathbb{E}\left[\|X_n - X(t_n)\|\right] = 0.$$

Moreover, we say that its *strong order of convergence* is p if there exist two positive numbers C and $\Delta t^\star$ such that

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \mathbb{E}\left[\|X_n - X(t_n)\|\right] \leq C \Delta t^p, \quad \text{for any } \Delta t \leq \Delta t^\star. \tag{9.26}$$

It is worth observing that, in many practical situations, an integer p satisfying (9.26) may not exist: especially for low regularity problems (such as some stochastic partial differential equations) an order $p - \varepsilon$ can be reached, for $\varepsilon > 0$.

An alternative notion of convergence is given as follows (see, for instance, [209, 213, 239] and the references therein).

Definition 9.4 With reference to the discretization (9.17), a numerical method for SDEs (9.7) providing the numerical solution $\{X_n\}_{n=0}^{L}$ is *weakly convergent* if

$$\lim_{\Delta t \to 0} \sup_{t_n \in \mathcal{I}_{\Delta t}} \left|\mathbb{E}\left[\Phi(X_n)\right] - \mathbb{E}\left[\Phi(X(t_n))\right]\right| = 0$$

for any test function Φ belonging to a suitable space S of functions. Moreover, we say that its *weak order of convergence* is p if there exist two positive numbers C and $\Delta t^\star$ such that

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \left|\mathbb{E}\big[\Phi(X_n)\big] - \mathbb{E}\left[\Phi(X(t_n))\right]\right| \leq C \Delta t^p, \quad \text{for any } \Delta t \leq \Delta t^\star.$$

In many practical situations, a suitable choice for the functional space S is given by the space of algebraic polynomials up to a given degree. Let us highlight the connection between the two above given notions of convergence. If we assume that $\Phi(x) = x$, a strongly convergent method is also weakly convergent. Indeed,

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \left| \mathbb{E}\left[\Phi(X_n)\right] - \mathbb{E}\left[\Phi(X(t_n))\right] \right| = \sup_{t_n \in \mathcal{I}_{\Delta t}} \left| \mathbb{E}\left[X_n\right] - \mathbb{E}\left[X(t_n)\right] \right|.$$

Since

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \left| \mathbb{E}\left[X_n\right] - \mathbb{E}\left[X(t_n)\right] \right| = \sup_{t_n \in \mathcal{I}_{\Delta t}} \left| \mathbb{E}\left[X_n - X(t_n)\right] \right| \leq \sup_{t_n \in \mathcal{I}_{\Delta t}} \mathbb{E}\Big[\|X_n - X(t_n)\|\Big],$$

we obtain

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \left| \mathbb{E}\left[\Phi(X_n)\right] - \mathbb{E}\left[\Phi(X(t_n))\right] \right| \leq \sup_{t_n \in \mathcal{I}_{\Delta t}} \mathbb{E}\Big[\|X_n - X(t_n)\|\Big],$$

as we aimed to prove.

Let us now give a proof of strong convergence for Euler-Maruyama method (9.18), following [213]. We observe that the proof relies on the previous knowledge of the following famous inequalities:

- Lyapunov inequality. For a given random variable Y,

$$\mathbb{E}\left[\|Y\|\right] \leq \sqrt{\mathbb{E}\left[\|Y\|^2\right]}; \tag{9.27}$$

- Cauchy-Schwarz inequality

$$\mathbb{E}\left[\left(\int_0^T b(r)\mathrm{d}r\right)^2\right] \leq T \int_0^T \mathbb{E}\left[b(r)^2\right]\mathrm{d}r. \tag{9.28}$$

Theorem 9.2 *Euler-Maruyama method* (9.18) *is strongly convergent with strong order* $\frac{1}{2}$.

Proof For the sake of simplicity of the presentation, let us provide the proof for scalar SDEs (9.7). We aim to prove the existence of a positive C such that

$$\sup_{t_n \in \mathcal{I}_{\Delta t}} \mathbb{E}\left[|X_n - X(t_n)|\right] \leq C\sqrt{\Delta t},$$

with C independent on Δt. To this purpose, it is convenient to proceed along continuous-time approximations. In other terms, we extend the pointwise approximation given at each grid point $t_n \in \mathcal{I}_{\Delta t}$ to a continuous approximant defined for any $t \in [0, T]$.

We consider a piecewise-constant process

$$\bar{X}(t) = X_n, \quad t_n \le t \le t_{n+1}$$

and define

$$Z(t) = \sup_{0 \le s \le t} \mathbb{E}\left[\left(\bar{X}(s) - X(s)\right)^2\right].$$

If we show that there exists a positive γ independent of Δt such that $Z(t) \le \gamma \Delta t$ then, by Lyapunov inequality (9.27), the thesis follows since

$$\sup_{0 \le s \le t} \mathbb{E}\left[\left|\bar{X}(s) - X(s)\right|\right] \le \sqrt{\sup_{0 \le s \le t} \mathbb{E}\left[\left(\bar{X}(s) - X(s)\right)^2\right]} \le \sqrt{\gamma}\sqrt{\Delta t}.$$

Given any point $s \in [0, T]$, let us denote by n_s the integer number such that $s \in [t_{n_s}, t_{n_s+1})$. In other terms, $\bar{X}(s) = X_{n_s}$. Then,

$$\begin{aligned}\bar{X}(s) - X(s) &= X_{n_s} - X(s) \\ &= X_{n_s} - \left(X_0 + \int_0^s f(X(r))\mathrm{d}r + \int_0^s g(X(r))\mathrm{d}W(r)\right).\end{aligned}$$

Let us replace $X_{n_s} - X_0$ by the telescopic sum

$$\sum_{i=0}^{n_s-1} (X_{i+1} - X_i)$$

and write each difference in this sum by means of Euler-Maruyama method (9.18), leading to

$$\begin{aligned}\bar{X}(s) - X(s) = &\sum_{i=0}^{n_s-1} f(X_i)\Delta t + \sum_{i=0}^{n_s-1} g(X_i)\Delta W_{i+1} \\ &- \int_0^s f(X(r))\mathrm{d}r - \int_0^s g(X(r))\mathrm{d}W(r).\end{aligned}$$

By construction,

$$\int_{t_i}^{t_{i+1}} f(\bar{X}(t))\mathrm{d}t = \int_{t_i}^{t_{i+1}} f(X_i)\mathrm{d}t = f(X_i)\Delta t$$

and, proceeding in similar way,

$$\int_{t_i}^{t_{i+1}} g(\bar{X}(t))\mathrm{d}W(t) = g(X_i)\Delta W_{i+1}.$$

Then,

$$\bar{X}(s) - X(s) = \int_0^{t_{n_s}} f(\bar{X}(r))\mathrm{d}r + \int_0^{t_{n_s}} g(\bar{X}(r))\mathrm{d}W(r) - \int_0^s f(X(r))\mathrm{d}r - \int_0^s g(X(r))\mathrm{d}W(r)$$

and, since $s \in [t_{n_s}, t_{n_s+1})$, we have

$$\bar{X}(s) - X(s) = \int_0^{t_{n_s}} \left(f(\bar{X}(r)) - f(X(r))\right)\mathrm{d}r + \int_0^{t_{n_s}} \left(g(\bar{X}(r)) - g(X(r))\right)\mathrm{d}W(r) - \int_{t_{n_s}}^s f(X(r))\mathrm{d}r - \int_{t_{n_s}}^s g(X(r))\mathrm{d}W(r).$$

Since, for any $a, b, c, d \in \mathbb{R}$,

$$(a+b+c+d)^2 \leq 4(a^2+b^2+c^2+d^2),$$

squaring and passing to expected values lead to

$$\mathbb{E}\left[\left(\bar{X}(s) - X(s)\right)^2\right] \leq 4(A_1 + A_2 + A_3 + A_4), \tag{9.29}$$

where

$$A_1 = \mathbb{E}\left[\left(\int_0^{t_{n_s}} \left(f(\bar{X}(r)) - f(X(r))\right)\mathrm{d}r\right)^2\right],$$

$$A_2 = \mathbb{E}\left[\left(\int_0^{t_{n_s}} \left(g(\bar{X}(r)) - g(X(r))\right)\mathrm{d}W(r)\right)^2\right],$$

$$A_3 = \mathbb{E}\left[\left(\int_{t_{n_s}}^s f(X(r))\mathrm{d}r\right)^2\right], \quad A_4 = \mathbb{E}\left[\left(\int_{t_{n_s}}^s g(X(r))\mathrm{d}W(r)\right)^2\right].$$

Let us give a separate bound for each of these four terms.

- Estimate of A_1. It relies on Cauchy-Schwarz inequality (9.28) and the Lipschitz continuity of the drift f of the equation (according to Theorem 9.1 of existence and uniqueness of the solution of SDEs), namely

$$\begin{aligned} A_1 &\leq t_{n_s} \int_0^{t_{n_s}} \mathbb{E}\left[\left(f(\bar{X}(r)) - f(X(r))\right)^2\right] \mathrm{d}r \\ &\leq t_{n_s} K^2 \int_0^{t_{n_s}} \mathbb{E}\left[\left(\bar{X}(r) - X(r)\right)^2\right] \mathrm{d}r \\ &\leq t_{n_s} K^2 \int_0^{s} Z(r) \mathrm{d}r; \end{aligned}$$

- estimate of A_2. It relies on Itô isometry (9.5) and the Lipschitz continuity of the diffusion g of the equation, namely

$$\begin{aligned} A_2 &= \int_0^{t_{n_s}} \mathbb{E}\left[\left(g(\bar{X}(r)) - g(X(r))\right)^2\right] \mathrm{d}r \leq K^2 \int_0^{t_{n_s}} \mathbb{E}\left[\left(\bar{X}(r) - X(r)\right)^2\right] \mathrm{d}r \\ &\leq K^2 \int_0^{s} Z(r) \mathrm{d}r; \end{aligned}$$

- estimate of A_3. It relies on Cauchy-Schwarz inequality (9.28) and the linear growth condition (9.15), as follows:

$$A_3 \leq (s - t_{n_s}) \int_{t_{n_s}}^{s} \mathbb{E}\left[f(X(r))^2\right] \mathrm{d}r \leq \Delta t \Lambda \int_{t_{n_s}}^{s} (1 + \mathbb{E}\left[X(r)^2\right]) \mathrm{d}r.$$

  Supposing that $\mathbb{E}\left[(X(r)^2\right]$ is bounded and denoted an upper bound for

$$\Lambda \int_{t_{n_s}}^{s} \mathbb{E}\left[X(r)^2\right] \mathrm{d}r$$

  by C_1, we obtain

$$A_3 \leq \Delta t^2 C_1;$$

- estimate of A_4. It also relies on Itô isometry (9.5) and the linear growth condition (9.15), as follows:

$$A_4 = \int_{t_{n_s}}^{s} \mathbb{E}\left[g(X(r))^2\right] \mathrm{d}r \leq \Lambda \int_{t_{n_s}}^{s} \left(1 + \mathbb{E}\left[X(r)^2\right]\right) \mathrm{d}r \leq \Delta t C_1.$$

We are now able to finalize the estimate in (9.29), as follows:

$$\mathbb{E}\left[\left(\bar{X}(s) - X(s)\right)^2\right] \leq 4\left[(T+1)K^2 \int_0^s Z(r)\mathrm{d}r + C_1 \Delta t (T+1)\right]$$

or, equivalently,

$$Z(t) \leq B_1 \int_0^t Z(r)\mathrm{d}r + B_2 \Delta t$$

with $B_1 = 4(T+1)K^2$ and $B_2 = C_1(T+1)$. Then, by Grönwall lemma 1.1,

$$Z(t) \leq B_2 \mathrm{e}^{B_1 t} \Delta t,$$

leading to the thesis, with $\gamma = B_2 \mathrm{e}^{B_1 t}$. □

We observe that the final error estimate gained in Theorem 9.2 reveals possible drawbacks leading to a corruption of the overall accuracy of the scheme. Indeed, the provided error constants depend on the length of the time window and on the Lipschitz constants of the drift and diffusion of the problem. As a consequence, the method cannot result so accurate on sufficiently long time windows, as well as for problems with too large Lipschitz constants (as it happens for deterministic stiff problems). The study of long-term properties of stochastic numerical methods will specifically be addressed in Sect. 9.7.

The following result on weak convergence of Euler-Maruyama method (9.18) holds true. Its proof is here omitted, but the reader can find it in [213].

Theorem 9.3 *Euler-Maruyama method* (9.18) *is weak convergent with weak order 1.*

It is also possible to prove what follows (see, for instance, [37, 168, 208, 213, 303] and references therein):

- strong and weak orders of convergence of Milstein method (9.19) are both equal to 1. Then, Milstein method is an improvement of Euler-Maruyama method in the sense of strong convergence;
- ϑ-Maruyama methods (9.20) share the same strong and weak orders of convergence of Euler-Maruyama method (9.18), i.e., the strong order is equal to $1/2$ and the weak order is equal to 1;
- also ϑ-Milstein methods (9.23) share the same strong and weak orders of convergence of the underlying Milstein method (9.19), i.e., both strong and weak orders are equal to 1;

- the convergence of explicit SRK methods (9.24)–(9.25) relies on the convergence of the underlying deterministic RK methods plus an additional condition, since we need to have

$$\sum_{i=1}^{s} b_i = \sum_{i=1}^{s} q_i = 1.$$

Example 9.6 Let us provide an experimental check of the strong orders of convergence of Euler-Maruyama method (9.18), the stochastic trapezoidal method (9.21) and the implicit Euler-Maruyama method (9.22) for the numerical solution of the geometric Brownian motion equation (9.9).

We repeatedly use Program 9.5 to sample of certain number of numerical trajectories useful to provide an estimate to the expected value needed for the computation of the strong order. Indeed, this is the idea of the so-called *Monte Carlo estimate* of the mean: from the Strong Law of Large Numbers, we know that repeated samples from a random variable can be averaged to give an asymptotically correct estimate of its mean. If Y is a random variable, a Monte Carlo estimate of its expectation can be computed through the following recipe:

- take a large number M of samples of Y. Denote them by $\{\xi_i\}_{i=1}^{M}$;
- compute the arithmetic mean (also known as *sample mean*)

$$a_M := \frac{1}{M} \sum_{i=1}^{M} \xi_i.$$

From the Central Limit Theorem, the discrepancy between sampled and true means depends on $b^2 := \mathrm{Var}[Y]$. Moreover, the estimate is only asymptotically correct (i.e., for large values of M). More details on the topic are given, for instance, in [213].

Table 9.2 provides the expected strong error in the final integration point associated to the application of the aforementioned methods, together with an estimate of the strong order, computed by a formula analogous to (3.23). For each value of the stepsize, strong errors are estimated by sampled means computed over $M = 1000$ trajectories of the numerical solutions. The theoretical results on the strong convergence are recovered by the numerical evidence, as visible from the table.

Table 9.2 Example 9.6: expected strong error in the final integration point associated to the application of the ϑ-Maruyama methods (9.20) with $\vartheta = 0, 1/2, 1$ to the equation of the geometric Brownian motion (9.9). An estimate of the strong order is also listed for each considered case. For each value of the stepsize, strong errors are estimated by sampled means computed over $M = 1000$ trajectories of the numerical solutions, each computed with $N = 2^8$ Wiener points and $L = N/R$ grid points, with R displayed in the table

R	$\text{err}_{\text{STRONG}}$	p_{STRONG}
$\vartheta = 0$		
8	$7.83 \cdot 10^{-2}$	
4	$5.06 \cdot 10^{-2}$	0.63
2	$3.54 \cdot 10^{-2}$	0.51
1	$2.47 \cdot 10^{-2}$	0.52
$\vartheta = 1/2$		
8	$7.11 \cdot 10^{-2}$	
4	$4.82 \cdot 10^{-2}$	0.56
2	$3.46 \cdot 10^{-2}$	0.48
1	$2.44 \cdot 10^{-2}$	0.50
$\vartheta = 1$		
8	$8.06 \cdot 10^{-2}$	
4	$5.34 \cdot 10^{-2}$	0.59
2	$3.54 \cdot 10^{-2}$	0.59
1	$2.55 \cdot 10^{-2}$	0.48

9.6 Linear Stability Analysis

We conclude this chapter by analyzing the linear stability properties of the numerical methods for SDEs (9.7) introduced in the previous sections, trying to extend the qualitative principles studied in the deterministic case (see Chap. 6) to the stochastic one [37, 89, 208, 209, 213, 306].

As in the deterministic case, the first step in performing a linear stability analysis requires providing a proper test problem, whose qualitative and quantitative features may be well detected and clarified. As proposed by Saito and Mitsui in [306], the test problem we consider is the following *stochastic Dahlquist test problem* given by the equation for the geometric Brownian motion

$$\begin{cases} \mathrm{d}X(t) = \mu X(t)\,\mathrm{d}t + \sigma X(t)\,\mathrm{d}W(t), \quad t \in [0, T], \\ X(0) = X_0, \end{cases} \tag{9.30}$$

with complex parameters μ and σ. Equation (9.30) is scalar and linear (both in the drift and in the diffusion) and for $\sigma = 0$ it recovers the deterministic Dahlquist test problem (6.1).

9.6.1 Mean-Square Stability

We give the following fundamental definition and a subsequent characterizing theorem.

Definition 9.5 The solution $X(t)$ of the stochastic Dahlquist test problem (9.30) is said to be *mean-square stable* if

$$\lim_{t\to\infty} \mathbb{E}\left[\|X(t)\|^2\right] = 0.$$

Theorem 9.4 *The solution $X(t)$ of the stochastic Dahlquist test problem (9.30) is mean-square stable if and only if*

$$\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2 < 0.$$

Proof Let us apply Itô formula to the quadratic function $u(X(t)) = X(t)^2$, where $X(t)$ is solution to (9.30), leading to

$$u(X(t)) = u(X_0) + 2\left(\mu + \frac{1}{2}|\sigma|^2\right)\int_0^t u(s)\mathrm{d}s + 2\sigma\int_0^t u(s)\mathrm{d}W(s).$$

Passing to side-by-side expectation leads to the linear SDE

$$\mathbb{E}\left[u(X(t))\right] = u(X_0) + 2\left(\mu + \frac{1}{2}|\sigma|^2\right)\int_0^t \mathbb{E}\left[u(X(s))\right]\mathrm{d}s$$

and, denoting by $y(t) = \mathbb{E}\left[u(X(t))\right]$, this equation is equivalent to the linear ODE

$$y'(t) = 2\left(\mu + \frac{1}{2}|\sigma|^2\right)y(t)$$

whose solution is given by

$$y(t) = \mathrm{e}^{2(\mu+\frac{1}{2}|\sigma|^2)t}\, y(0).$$

Then,

$$\mathbb{E}\left[|X(t)|^2\right] = \mathrm{e}^{2(\mu+\frac{1}{2}|\sigma|^2)t}|X_0|^2,$$

where X_0 is assumed to be deterministic and the thesis holds true. □

It is now natural asking when a numerical method for SDEs preserve the mean-square character of the solution along the discretized dynamics, according to the following definition.

Definition 9.6 A numerical method that employs the set of grid points (9.17) and provides the approximate solution $\{X_n\}_{n=0}^L$ to the linear scalar test SDE (9.30) is said to be *mean-square stable* if

$$\lim_{n\to\infty} \mathbb{E}\left[|X_n|^2\right] = 0.$$

Example 9.7 Let us analyze the mean-square stability of Euler-Maruyama method (9.18). To this purpose, let us apply this method to the stochastic Dahlquist test problem (9.30) with real parameters

$$X_{n+1} = (1 + \mu\Delta t)X_n + \sigma X_n \Delta W_{n+1}$$

and square it side-by-side so as to get

$$X_{n+1}^2 = (1 + \mu\Delta t)^2 X_n^2 + 2(1 + \mu\Delta t)\sigma X_n^2 \Delta W_{n+1} + \sigma^2 X_n^2 \Delta W_{n+1}^2.$$

Let us pass to side-by-side expectation, taking into account that

$$\mathbb{E}[\Delta W_{n+1}] = 0, \quad \mathbb{E}[\Delta W_{n+1}^2] = \Delta t$$

and that X_n and ΔW_{n+1} are independent random variables, leading to

$$\mathbb{E}[X_{n+1}^2] = \left((1 + \mu\Delta t)^2 + \sigma^2\Delta t\right)\mathbb{E}[X_n^2].$$

In other terms, Euler-Maruyama method is mean-square stable if

$$(1 + \mu\Delta t)^2 + \sigma^2\Delta t < 1.$$

(continued)

Example 9.7 (continued)
As in the deterministic case, the stability of the numerical solution provided by an explicit method requires fulfilling a stepsize restriction. In Fig. 9.6, the stability region of the problem (9.30)

$$S_{SDE} = \left\{ \lambda, \mu \in \mathbb{C} : \mathrm{Re}(\lambda) + \frac{1}{2}|\mu|^2 < 0 \right\} \tag{9.31}$$

and that of Euler-Maruyama method (9.18)

$$S_{EM} = \left\{ \lambda, \mu \in \mathbb{R} : (1 + \mu\Delta t)^2 + \sigma^2 \Delta t < 1 \right\} \tag{9.32}$$

are depicted and compared. The fact that the stability region of the method is much smaller than that of the problem is a signal of the possible stepsize restrictions affecting the efficiency of the method; as we are going to explain, stochastic ϑ-methods significantly improve the stability of Euler-Maruyama method.

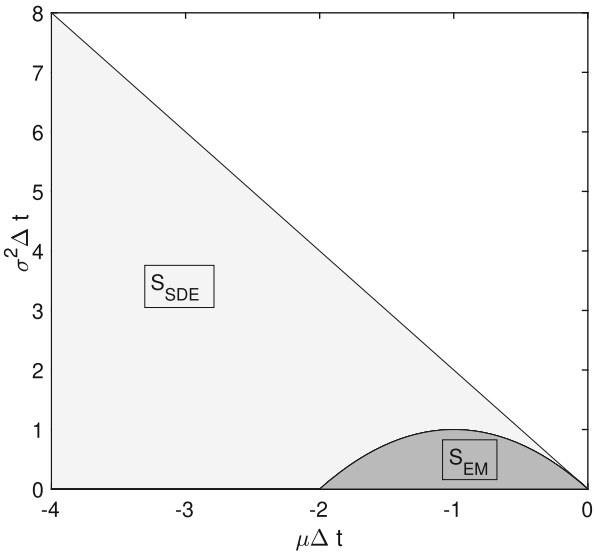


Fig. 9.6 Region of mean-square stability (9.31) of the solution to the stochastic Dahlquist test problem (9.30) (light grey), for real values of the parameters μ and σ, vs the mean-square stability region (9.32) of Euler-Maruyama method (9.18) (dark grey)

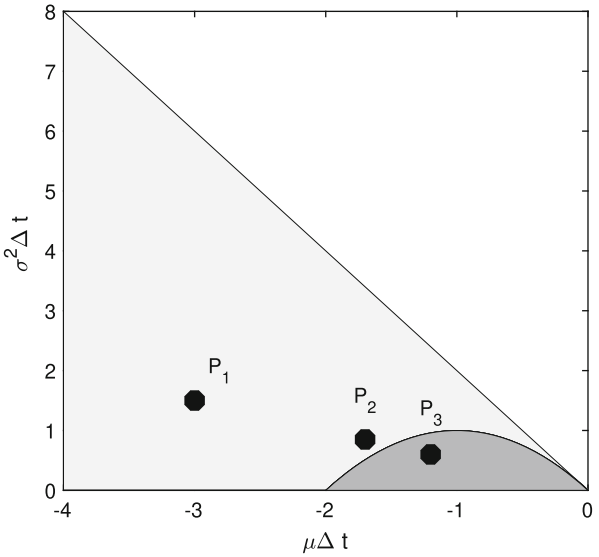


Fig. 9.7 Example 9.8: points chosen for the numerical experiments. P_1 and P_2 lie outside the stability region of the method (P_2 is close to its boundary), but inside the stability region of the problem; P_3 lies in both regions

Example 9.8 With reference to Fig. 9.6, let us choose the following 3 points

$$P_1 = \left(-3, \frac{3}{2}\right), \quad P_2 = \left(-\frac{9}{5}, \frac{9}{10}\right), \quad P_3 = \left(-\frac{6}{5}, \frac{3}{5}\right),$$

highlighted in Fig. 9.7. P_1 and P_2 lie outside the stability region of the method (P_2 is close to its boundary), but inside the stability region of the problem; P_3 lies in both regions. If we consider $\mu = -4$ and $\sigma = \sqrt{2}$, the values of Δt respectively corresponding to these points are

$$\Delta t_1 = \frac{3}{4}, \quad \Delta t_2 = \frac{9}{20}, \quad \Delta t_3 = \frac{3}{10}.$$

Let us apply Euler-Maruyama method (9.18) to the stochastic Dahlquist test problem (9.30) with $\mu = -4$ and $\sigma = \sqrt{2}$, using above stepsize values. The corresponding numerical evidence is reported in Fig. 9.8, where the Monte Carlo estimates of the mean-squares are computed over $M = 1000$ Euler-Maruyama paths. Numerical results confirm the theoretical results: the mean-square associated to Δt_1 blows up, in coherence with the fact that the corresponding point only lies inside the stability region of the problem and outside that of the numerical method; the mean-square associated to Δt_3 exponentially decreases (be aware that the graph is in semi-logarithmic scale), in coherence with the fact that the corresponding point lies inside both the

(continued)

Example 9.8 (continued)
stability region of the problem and that of the numerical method; the mean-square associated to Δt_2 does not blow up or exponentially decrease, since the corresponding point lies close to the boundary of the mean-square stability region of the numerical method, though outside it.

9.6.2 Mean-Square Stability of Stochastic ϑ-Methods

Let us now analyze the mean-square stability properties of stochastic ϑ-methods, according to the results provided in [208, 213]. We first consider ϑ-Maruyama methods (9.20), as follows.

Theorem 9.5 *The mean-square stability region of ϑ-Maruyama methods (9.20) is given by*

$$S_\vartheta = \left\{ \mu, \sigma \in \mathbb{C} : (1 - 2\vartheta)|\mu|^2 \Delta t < -2\left(\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2 \right) \right\}.$$

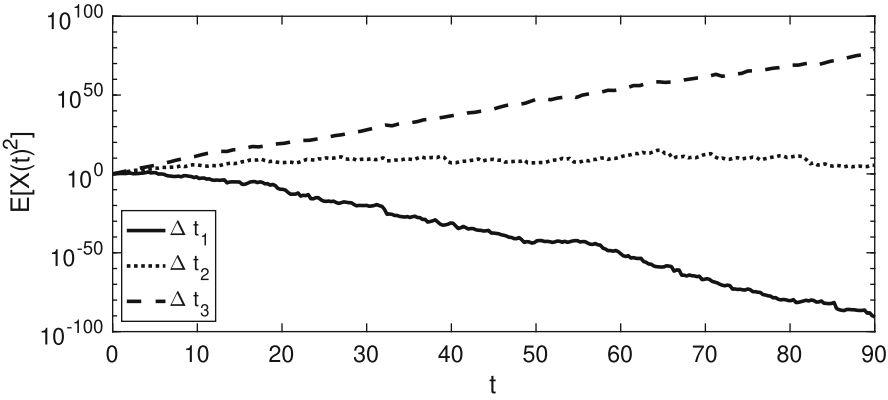


Fig. 9.8 Mean-square of the approximate solution to the stochastic Dahlquist test problem (9.30) with $\mu = -4$ and $\sigma = \sqrt{2}$, arisen from the application of Euler-Maruyama method (9.18) with $\Delta t_1 = \frac{3}{4}$ (dashed line), $\Delta t_2 = \frac{9}{20}$ (dotted line) and $\Delta t_3 = \frac{3}{10}$ (solid line). Monte Carlo estimates of the mean-squares are computed over $M = 1000$ Euler-Maruyama paths

Proof Applying the ϑ-Maruyama method (9.20) to the stochastic Dahlquist test problem (9.30) yields

$$X_{n+1} = \frac{1 + (1-\vartheta)\mu\Delta t + \sigma\Delta W_{n+1}}{1 - \vartheta\Delta t\mu} X_n.$$

Side-by-side squaring and passing to expectation leads to

$$|1 - \vartheta\mu\Delta t|^2 \mathbb{E}\big[|X_{n+1}|^2\big] = \mathbb{E}\big[|1 + (1-\vartheta)\mu\Delta t + \sigma\Delta W_n|^2\big]\mathbb{E}\big[|X_n|^2\big].$$

Hence, we have obtained the recurrence relation

$$\mathbb{E}\big[|X_{n+1}|^2\big] = \gamma(\vartheta, \Delta t)\mathbb{E}\big[|X_n|^2\big],$$

where

$$\gamma(\vartheta, \Delta t) = \frac{1 + (1-\vartheta)^2\Delta t^2|\mu|^2 + \sigma|^2\Delta t + 2(1-\vartheta)\mathrm{Re}(\mu)\Delta t}{1 + \vartheta^2|\mu|^2\Delta t^2 - 2\vartheta\mathrm{Re}(\mu)\Delta t}.$$

Then, mean-square stability holds true if and only if $\gamma(\vartheta, \Delta t) < 1$, that is equivalent to

$$(1 - 2\vartheta)|\mu|^2\Delta t < -2\left(\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2\right), \tag{9.33}$$

leading to the thesis. □

In analyzing the inequality in (9.33), always take into account that, for mean-square stable SDEs, $\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2 < 0$. Let us observe that mean-square stability condition (9.33) for ϑ-Maruyama methods (9.20) allows us to conclude what follows: for any $\Delta t > 0$,

- if $0 \le \vartheta < \frac{1}{2}$, the stability region S_ϑ of the method is a subset of the stability region S_{SDE} (9.31) of the problem. For any $(\mu, \sigma) \in S_{SDE}$, the method is mean-square stable if and only if

$$\Delta t < -\frac{2\left(\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2\right)}{(1 - 2\vartheta)|\mu|^2},$$

  while for unstable SDEs, the method is not stable, for any choice of $\Delta t > 0$. In other terms, the mean-square stability region of ϑ-Maruyama methods with $0 \le \vartheta < \frac{1}{2}$ is bounded and the method is stable subject to stepsize restrictions;
- if $\vartheta = \frac{1}{2}$, the stochastic trapezoidal method (9.21) is recovered and its stability region coincides with that of the problem, giving a mean-square generalization of the concept of A-stability, that we can denote as *mean-square A-stability*. So,

for a mean-square stable SDE, the stochastic trapezoidal method is mean-square stable for any $\Delta t > 0$;

- if $\frac{1}{2} < \vartheta \leq 1$, the stability region of the method contains that of the problem and the method is stable for any choice of the stepsize, when applied to a mean-square stable SDE. Actually, these methods are also *overstable* since, when applied to unstable problems, they are not mean-square stable for any

$$\Delta t < -\frac{2\left(\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2\right)}{(1-2\vartheta)|\mu|^2},$$

so they can also be stable on an unstable SDE.

Let us now move to the case of stochastic ϑ-Milstein methods (9.23), whose linear stability analysis has been provided in [117]. Applying (9.23) to the stochastic Dahlquist test problem (9.30) yields

$$X_{n+1} = X_n + (1-\vartheta)\mu\Delta t X_n + \vartheta\mu\Delta t X_{n+1} + \sigma X_n \Delta W_{n+1} + \frac{\sigma^2}{2} X_n (\Delta W_{n+1}^2 - \Delta t),$$

i.e.,

$$(1-\vartheta\mu\Delta t)X_{n+1} = \left(1 + (1-\vartheta)\mu\Delta t + \sigma\Delta W_{n+1} + \frac{\sigma^2}{2}(\Delta W_{n+1}^2 - \Delta t)\right) X_n.$$

It is convenient to write

$$1 + (1-\vartheta)\mu\Delta t + \sigma\Delta W_n + \frac{1}{2}\sigma^2(\Delta W_n^2 - \Delta t) = x + iy,$$

with

$$x = 1 + (1-\vartheta)\mathrm{Re}(\mu)\Delta t + \mathrm{Re}(\sigma)\Delta W_n + \frac{1}{2}\mathrm{Re}\big(\sigma^2\big)(\Delta W_n^2 - \Delta t),$$

$$y = (1-\vartheta)I(\mu) + \mathrm{Im}(\sigma)\Delta W_n + \frac{1}{2}\mathrm{Im}\big(\sigma^2\big)(\Delta W_n^2 - \Delta t).$$

The expected value of $x + iy$ is given by

$$\mathbb{E}\big[|x+iy|^2\big] = \mathbb{E}\big[x^2 + y^2\big] = 1 + (1-\vartheta)^2\Delta t^2|\mu|^2 + \Delta W_n^2|\sigma|^2 + \frac{1}{4}(\Delta W_n^2 - \Delta t)^2|\sigma^2|^2 + 2(1-\vartheta)\mathrm{Re}(\mu)\Delta t$$

and, as a consequence,

$$\mathbb{E}\big[|X_{n+1}|^2\big] = \beta(\vartheta, \Delta t)\mathbb{E}\big[|X_n|^2\big],$$

with

$$\beta(\vartheta, \Delta t) = \frac{1 + (1-\vartheta)^2 \Delta t^2 |\mu|^2 + \Delta t |\sigma|^2 + \frac{1}{2}\Delta t^2 |\sigma^2|^2 + 2(1-\vartheta)\mathrm{Re}(\mu)\Delta t}{1 + \vartheta^2 |\mu|^2 \Delta t^2 - 2\vartheta \mathrm{Re}(\mu)\Delta t}.$$

In other terms, ϑ-Milstein methods (9.23) provide a mean-square stable numerical solution if and only if

$$\big((1-2\vartheta)|\mu|^2 + \frac{1}{2}|\sigma^2|^2\big)\Delta t < -2\big(\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2\big).$$

For mean-square stable test problem (9.30), the right-hand side of last inequality is always positive. Then, for values of $\vartheta \in [0, 1]$ such that

$$(1-2\vartheta)|\mu|^2 + \frac{1}{2}|\sigma^2|^2 < 0, \tag{9.34}$$

any numerical solution computed by (9.23) is mean-square stable for any choice of the stepsize Δt. Condition (9.34) is equivalent to

$$\vartheta > \frac{1}{2} + \frac{|\sigma^2|^2}{4|\mu|^2}, \quad 0 \le \vartheta \le 1. \tag{9.35}$$

Two situations may occur:

- if the right-hand side of (9.35) is greater than 1, then no ϑ-Milstein method with $\vartheta \in [0, 1]$ is mean-square stable for any choice of Δt;
- if the right-hand side of (9.35) is smaller than 1 then, for any

  $$0 \le \vartheta \le 1/2 + \big(|\sigma^2|^2\big)/\big(4|\mu|^2\big),$$

  the corresponding ϑ-Milstein method provides a mean-square stable numerical solution to (9.30) subject to the stepsize restriction

  $$\Delta t < \frac{2\big|\mathrm{Re}(\mu) + \frac{1}{2}|\sigma|^2\big|}{(1-2\vartheta)|\mu|^2 + \frac{1}{2}|\sigma^2|^2}.$$

To conclude, while ϑ-Maruyama methods allow a large variety of mean-square A-stable methods, this is not the case of ϑ-Milstein methods, since stepsize restrictions have to be satisfied in order to provide mean-square stable ϑ-Milstein numerical solutions.

9.6.3 A-stability Preserving SRK Methods

We now focus on analyzing the mean-square stability properties of stochastic Runge-Kutta methods (9.24). We first present the following result [89].

Theorem 9.6 *If $\{X_n\}_{n=0}^{L}$ is the numerical solution of (9.30) computed by a SRK method (9.24), with reference to the set of grid points (9.17), then the following recurrence relation holds true*

$$\mathbb{E}\left[|X_{n+1}|^2\right] = R_s(\eta, \zeta)\mathbb{E}\left[|X_n|^2\right],$$

where

$$R_s(\eta, \zeta) = \left(1 + |\zeta|^2\right)|R_d(\eta)|^2, \tag{9.36}$$

being $R_d(\eta) = 1 + \eta b^{\intercal}(I - \eta A)^{-1}e$ the stability function (6.9) of the underlying deterministic RK method (4.8), $I \in \mathbb{R}^{s\times s}$ the identity matrix and $e \in \mathbb{R}^s$ the unit vector.

Proof Applying the SRK method (9.24) to the stochastic Dahlquist test problem (9.30) gives the following recurrence

$$\begin{cases} \widehat{X}_i = X_n + \mu\Delta t \sum_{j=1}^{s} a_{ij}\widehat{X}_j + \sigma\Delta W_{n+1}X_n, \quad i = 1, 2, \ldots, s, \\ X_{n+1} = X_n + \mu\Delta t \sum_{i=1}^{s} b_i\widehat{X}_i + \sigma\Delta W_{n+1}X_n, \end{cases}$$

that we can recast in the compact form

$$\begin{cases} \widehat{X} = X_n e + \eta A\widehat{X} + \zeta\xi_n X_n e, \\ X_{n+1} = X_n + \eta b^{\intercal}\widehat{X} + \zeta\xi_n X_n, \end{cases} \tag{9.37}$$

with $\widehat{X} = [\widehat{X}_1, \ldots, \widehat{X}_s]^{\intercal}$, $\eta = \mu\Delta t$ and with $\zeta = \sigma\sqrt{\Delta t}$, being ξ_n a standard normal random variable. Manipulating the first equation in (9.37) yields

$$\widehat{X} = (I - \eta A)^{-1}e(1 + \zeta\xi_n)X_n$$

and inserting this expression of $\widehat{X}$ into the second equation in (9.37) leads to

$$\begin{aligned} X_{n+1} &= X_n + \eta b^{\intercal}\left[(I-\eta A)^{-1} e\,(1+\zeta\xi_n)X_n\right] + \zeta\xi_n X_n \\ &= \left(1 + \eta b^{\intercal}\left[(I-\eta A)^{-1} e(1+\zeta\xi_n)\right] + \zeta\xi_n\right) X_n \\ &= \left[1 + \eta b^{\intercal}(I-\eta A)^{-1} e + \eta b^{\intercal}(I-\eta A)^{-1} e\zeta\xi_n + \zeta\xi_n\right] X_n \\ &= \left[1 + \eta b^{\intercal}(I-\eta A)^{-1} e + \zeta\xi_n(1 + \eta b^{\intercal}(I-\eta A)^{-1} e)\right] X_n \\ &= (1+\zeta\xi_n) R_d(\eta) X_n. \end{aligned}$$

Since $|1+\zeta\xi_n|^2 = 1 + |\zeta|^2\xi_n^2$, side-by-side squaring and passing to expectation yields

$$\mathbb{E}\left[|X_{+1}|^2\right] = \mathbb{E}\left[1+|\zeta|^2\xi_n^2\right]|R_d(\eta)|^2\mathbb{E}\left[|X_n|^2\right] = \left(1+|\zeta|^2\right)|R_d(\eta)|^2\mathbb{E}\left[|X_n|^2\right].$$

The definition of $R_s(\eta,\zeta)$ gives the thesis. □

Definition 9.7 The function $R_s(\eta,\zeta)$ defined in (9.36) is called *mean-square stability function* of the SRK method (9.24).

Definition 9.8 A SRK method (9.24) is *mean-square stable* for a fixed couple $(\eta,\zeta) \in \mathbb{C}^2$, if

$$R_s(\eta,\zeta) < 1, \tag{9.38}$$

with $R_s(\eta,\zeta)$ defined in (9.36). Moreover, the set

$$S_{\mathrm{SRK}} = \{(\eta,\zeta) \in \mathbb{C}^2 : R_s(\eta,\zeta) < 1\}$$

is called *mean-square stability region* of the SRK method (9.24).

Definition 9.9 A SRK method (9.24) is said to be *mean-square A-stable* if

$$S_{\mathrm{SRK}} \supseteq S_{\mathrm{SDE}},$$

being S_{SDE} the region of mean-square stability (9.31) of the stochastic Dahlquist test problem (9.30).

We can appreciate from Theorem 9.6 that the stability function of a SRK method depends on the stability function of the underlying deterministic RK method. So, there is a deep link between the stability properties of SRK methods (9.24) arising as perturbation of deterministic RK methods (4.8) and the latter. It is then natural to ask if the properties of the underlying deterministic RK method are inherited by the stochastic perturbation leading to (9.24). This aspect is clarified by the following result [89].

Theorem 9.7 *For a given A-stable deterministic Runge-Kutta method* (4.8)*, the corresponding stochastic perturbation* (9.24) *is mean-square A-stable if and only if*

$$|R_d(\eta)|^2 \leq \frac{1}{1-2\mathrm{Re}(\eta)}, \quad \text{for any } \eta \in \mathbb{C}^-. \tag{9.39}$$

Proof For a SRK method (9.24), whose underlying deterministic RK method (4.8) is A-stable, taking into account the definition of stability function as in (9.31), we obtain from the condition (9.38) that $\left(1+|\zeta|^2\right)|R_d(\eta)|^2 < 1$, i.e.,

$$|\zeta|^2 < \frac{1}{|R_d(\eta)|^2} - 1. \tag{9.40}$$

We know from Definition 9.9 that the mean-square A-stability of the SRK method (9.24) is equivalent to the condition $R_s(\eta, \zeta) < 1$, for any $\eta, \zeta \in S_{\mathrm{SDE}}$. By definition itself, $\eta, \zeta \in S_{\mathrm{SDE}}$ if and only if

$$|\zeta|^2 \leq -2\mathrm{Re}(\eta). \tag{9.41}$$

Hence, taking into account (9.40) and (9.41), we have

$$\frac{1}{|R_d(\eta)|^2} - 1 \leq -2\mathrm{Re}(\eta),$$

i.e.,

$$|R_d(\eta)|^2 \leq \frac{1}{1-2\mathrm{Re}(\eta)},$$

concluding the proof. □

Example 9.9 The SRK method (9.24) having the one-stage Gaussian method (4.23) as underlying deterministic RK method does not inherit the A-stability property. Indeed, we recall that

$$R_d(\eta) = \frac{1+\frac{1}{2}\eta}{1-\frac{1}{2}\eta},$$

then (9.39) holds true only for $\mathrm{Re}(\eta) \geq -\frac{1}{4}|\eta|^2$.

Both the SRK methods (9.24) having the one-stage Radau IA and IIA methods (introduced in Sect. 4.4.2) as underlying RK ones are mean-square A-stable. Indeed, both deterministic methods share the same stability function

$$R_d(\eta) = \frac{1}{1-\eta}.$$

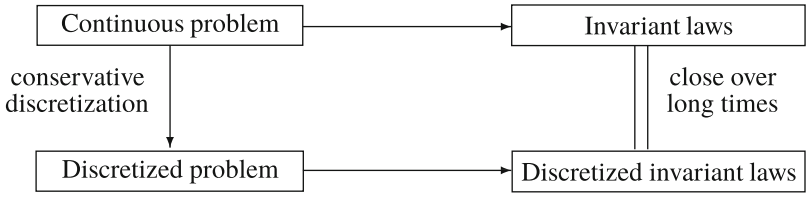
Then, condition (9.39) is equivalent to

$$\frac{1}{|1-\eta|^2} \leq \frac{1}{1-2\mathrm{Re}(\eta)},$$

that holds true for any $\eta \in \mathbb{C}$, as required by Theorem 9.7.

9.7 Principles of Stochastic Geometric Numerical Integration

Chapter 8 has fully been dedicated to deterministic geometric numerical integration, with the aim to understand how to maintain characteristic features of the continuous problem along the numerical dynamics. Even if stochastic dynamics is governed by random fluctuations, this chapter is devoted to provide examples of conservation issues along the approximate solutions of SDEs. Indeed, there is a principle of geometric numerical integration also for SDEs, briefly summarized in the following diagram.



Typical situations where invariant laws are characteristic properties of SDEs arise, for instance, in stochastic oscillators [51, 54, 88, 91, 93, 120, 127, 129, 130, 135, 142, 169, 175, 313, 330, 341], in nonlinear SDEs with one-sided Lipschitz drift leading to mean-square contractive dynamics [38, 117–119, 212, 215], in stochastic Hamiltonian problems [11, 12, 21, 47, 48, 85, 121, 136, 143, 217, 218, 247, 256, 270–273, 333].

This section aims to provide selected examples of conservation of invariant laws of SDEs along their discretized dynamics.

9.7.1 Nonlinear Stability Analysis: Exponential Mean-Square Contractivity

Let us first present the following result, providing a stability inequality for nonlinear Itô SDEs (9.7) satisfying proper regularity assumptions. The proof is here omitted, but the interested reader can find it in [212].

Theorem 9.8 *For a given nonlinear SDE (9.7), let us assume the following properties for the drift f and the diffusion g:*

(i) $f, g \in C^1(\mathbb{R}^d)$;

(ii) *f satisfies a one-sided Lipschitz condition, i.e., there exists $\mu_f \in \mathbb{R}$ such that*

$$< x - y, f(x) - f(y) > \leq \mu_f \|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d;$$

(iii) *g is a globally Lipschitz function, i.e., there exists $L_g > 0$ such that*

$$\|g(x) - g(y)\|^2 \leq L_g \|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d.$$

(continued)

Theorem 9.8 (continued)
Then, any two solutions $X(t)$ and $Y(t)$ of (9.7), computed with respect to distinct initial values X_0 and Y_0 such that $\mathbb{E}\big[\|X_0\|^2\big] < \infty$ and $\mathbb{E}\big[\|Y_0\|^2\big] < \infty$, satisfy

$$\mathbb{E}\big[\|X(t) - Y(t)\|^2\big] \leq \mathbb{E}\big[\|X_0 - Y_0\|^2\big]\mathrm{e}^{\alpha t}, \tag{9.42}$$

where $\alpha = 2\mu_f + L_g$.

Equation (9.42) provides a measure of the gap between two solutions of the same SDE, computed in mean-square. Clearly, if the parameter α, appearing in (9.42) as rate of the exponential, is negative, then we can infer an exponential decay of the mean-square deviation between two solutions of a given SDE. Correspondingly, we give the following definition.

Definition 9.10 A nonlinear SDE (9.7) satisfying the inequality (9.42) with $\alpha < 0$ is said to generate *exponential mean-square contractive solutions*.

Let us observe that, when the diffusion g in (9.7) is identically equal to 0, Definition 9.10 recovers the classical contractivity condition $\mu_f < 0$ characterizing the deterministic case, according to the theory developed in Sects. 1.3 and 8.2.

We now aim to analyze under which conditions stochastic ϑ-methods (9.20) and stochastic Runge-Kutta methods (9.24) are able to reproduce the exponential mean-square contractive character given by Definition 9.10 along their numerical dynamics. The presentation relies on the results given in [117] for stochastic ϑ-methods and [118] for stochastic Runge-Kutta methods.

9.7.2 Mean-Square Contractivity of Stochastic ϑ-Methods

Let us start with a result reproducing the inequality (9.42) along the numerical dynamics of stochastic ϑ-Maruyama methods. This result requires a technical lemma, whose proof is here omitted, but the reader can find it in [215].

Lemma 9.1 *Under the assumptions (i)–(iii) given in Theorem 9.8, for any positive h and any $b_1, b_2 \in \mathbb{R}^d$, there exist unique $a_1, a_2 \in \mathbb{R}^d$, solutions of the implicit equations*

$$a_i - hf(a_i) = b_i, \quad i = 1, 2,$$

satisfying the inequality

$$(1 - 2h\mu_f)\|a_1 - a_2\|^2 \leq \|b_1 - b_2\|^2.$$

Theorem 9.9 *Under the assumptions (i)–(iii) given in Theorem 9.8, any two numerical solutions X_n and Y_n, $n \geq 0$, computed by applying the ϑ-Maruyama method (9.20) to (9.7) with distinct initial values X_0 and Y_0 such that $\mathbb{E}\left[|X_0|^2\right] < \infty$ and $\mathbb{E}\left[|Y_0|^2\right] < \infty$, satisfy the inequality*

$$\mathbb{E}\left[\|X_n - Y_n\|^2\right] \leq \mathbb{E}\left[\|X_0 - Y_0\|^2\right] \mathrm{e}^{\nu(\vartheta,\Delta t)t_n}, \tag{9.43}$$

where

$$\nu(\vartheta, \Delta t) = \frac{1}{\Delta t} \ln \beta(\vartheta, \Delta t) \tag{9.44}$$

and

$$\beta(\vartheta, \Delta t) = 1 + \frac{\alpha + (1 - \vartheta)^2 M_f \Delta t}{1 - 2\vartheta\mu_f \Delta t} \Delta t, \tag{9.45}$$

with

$$M_f = \sup_{t \in [0,T]} \mathbb{E}\left[\|f'(X(t))\|^2\right]. \tag{9.46}$$

Proof Applying (9.20) to (9.7) for the approximate computation of $X(t)$ and $Y(t)$ reads

$$X_{n+1} = X_n + (1 - \vartheta)\Delta t f(X_n) + \vartheta \Delta t f(X_{n+1}) + g(X_n)\Delta W_{n+1},$$

$$Y_{n+1} = Y_n + (1 - \vartheta)\Delta t f(Y_n) + \vartheta \Delta t f(Y_{n+1}) + g(Y_n)\Delta W_{n+1}.$$

Handling these implicit relations by means of Lemma 9.1 yields

$$(1 - 2\mu_f \vartheta \Delta t)\|X_{n+1} - Y_{n+1}\|^2 \leq \|X_n - Y_n + (1-\vartheta)\Delta t \Delta f_n + \Delta g_n \Delta W_{n+1}\|^2,$$

with $\Delta f_n = f(X_n) - f(Y_n)$ and $\Delta g_n = g(X_n) - g(Y_n)$.

Last inequality is then equivalent to

$$\begin{aligned}(1 - 2\mu_f \vartheta \Delta t)\|X_{n+1} - Y_{n+1}\|^2 \leq \|X_n - Y_n\|^2 + (1-\vartheta)^2 \Delta t^2 \|\Delta f_n\|^2 \\ + \|\Delta g_n \Delta W_{n+1}\|^2 \\ + 2(1-\vartheta)\Delta t \langle X_n - Y_n, \Delta f_n \rangle \\ + 2\langle X_n - Y_n, \Delta g_n \Delta W_{n+1} \rangle \\ + 2(1-\vartheta)\Delta t \langle \Delta f_n, \Delta g_n \Delta W_{n+1} \rangle .\end{aligned}$$

We pass to side-by-side expectation taking into account that

- Δg_n and ΔW_{n+1} are independent random variables. Then, assuming that we are using a matrix norm compatible with the vector norm, we have

$$\mathbb{E}\left[\|\Delta g_n \Delta W_{n+1}\|^2\right] \leq \mathbb{E}\left[\|\Delta g_n\|^2\right] \mathbb{E}\left[\|\Delta W_{n+1}\|^2\right]$$

  and by the Lipschitz continuity of g we obtain

$$\mathbb{E}\left[\|\Delta g_n \Delta W_{n+1}\|^2\right] \leq L_g \Delta t \, \mathbb{E}\left[\|X_n - Y_n\|^2\right];$$

- using the one-sided Lipschitz property satisfied by f, we have

$$\mathbb{E}\left[\langle X_n - Y_n, \Delta f_n \rangle\right] \leq \mu_f \, \mathbb{E}\left[\|X_n - Y_n\|^2\right];$$

- the expectation of W_{n+1} is zero;
- due to (9.46) and the regularity assumptions on f, we have

$$\mathbb{E}\left[\|\Delta f_n\|^2\right] \leq M_f \, \mathbb{E}\left[\|X_n - Y_n\|^2\right].$$

Then, we obtain

$$\mathbb{E}\big[\|X_{n+1} - Y_{n+1}\|^2\big] \leq \beta(\vartheta, \Delta t) \, \mathbb{E}\big[\|X_n - Y_n\|^2\big],$$

with $\beta(\vartheta, \Delta t)$ defined in (9.45). By recursion,

$$\mathbb{E}\big[\|X_n - Y_n\|^2\big] \leq \beta(\vartheta, \Delta t)^n \, \mathbb{E}\big[\|X_0 - Y_0\|^2\big].$$

Using (9.44), we have

$$\beta(\vartheta, \Delta t) = \mathrm{e}^{\Delta t\, \nu(\vartheta, \Delta t)},$$

leading to

$$\mathbb{E}\left[\|X_n - Y_n\|^2\right] \leq \mathrm{e}^{n\Delta t\, \nu(\vartheta, \Delta t)} \mathbb{E}\left[\|X_0 - Y_0\|^2\right] = \mathrm{e}^{\nu(\vartheta, \Delta t) t_n} \mathbb{E}\left[\|X_0 - Y_0\|^2\right],$$

giving the thesis. □

According to Theorem 9.9, stochastic ϑ-Maruyama methods (9.20), applied to a nonlinear SDE (9.7) whose solutions are mean-square contractive with parameter $\alpha = 2\mu_f + L_g$, are capable of reproducing the exponential mean-square inequality (9.42) with parameter $\nu(\vartheta, \Delta t)$ given by (9.44). Let us compute the gap between the exact rate α and its numerical counterpart $\nu(\vartheta, \Delta t)$ through the following result.

Theorem 9.10 *Under the same assumptions of Theorem 9.9, for any fixed value of* $\vartheta \in [0, 1]$*, we have*

$$|\nu(\vartheta, \Delta t) - \alpha| = O(\Delta t).$$

Proof Expanding $\nu(\vartheta, \Delta t)$ in (9.44) in power series of Δt yields

$$\nu(\vartheta, \Delta t) = \alpha + \left(M_f(\vartheta - 1)^2 - \frac{\alpha^2}{2} + 2\alpha\mu_f\vartheta\right)\Delta t + O(\Delta t^2),$$

leading to the thesis. □

Theorem 9.10 ensures that the numerical exponent $\nu(\vartheta, \Delta t)$ approaches the exact parameter α, when Δt tends to 0, as desirable. Let us also observe that, for the expansion of $\nu(\vartheta, \Delta t)$ in power series of Δt, we have used the symbolic framework of Matlab as follows:

```
>> syms x alf M mu th
>> f=log(1+(alf+(1-th)^2*M*x)*x/(1-2*th*mu*x))/x;
>> T=taylor(f,x);
>> coeffs(T,x)
```

A similar result can be given for ϑ-Milstein methods as follows. The proof, obtained by using similar arguments to those given for Theorem 9.9, is here omitted, but the reader can find it in [117].

Theorem 9.11 *Under the assumptions (i)–(iii) given in Theorem 9.8, any two numerical solutions X_n and Y_n, $n \geq 0$, computed by applying the ϑ-Milstein method*

$$X_{n+1} = X_n + (1-\vartheta)\Delta t f(t_n, X_n) + \vartheta \Delta t f(t_{n+1}, X_{n+1}) + \sum_{j=1}^{m} g^j(t_n, X_n)\Delta W_{n+1}^j + \frac{1}{2}\sum_{j=1}^{m} L^j g^j(t_n, X_n)\left(\Delta W_{n+1}^j{}^2 - \Delta t\right) + \frac{1}{2}\sum_{\substack{j_1, j_2=1 \\ j_1 \neq j_2}}^{m} L^{j_1} g^{j_2}(t_n, X_n)\Delta W_{n+1}^{j_1}\Delta W_{n+1}^{j_2} \tag{9.47}$$

to (9.7), with distinct initial values X_0 and Y_0 such that $\mathbb{E}\big[\|X_0\|^2\big] < \infty$ and $\mathbb{E}\big[\|Y_0\|^2\big] < \infty$, satisfy the inequality

$$\mathbb{E}\big[\|X_n - Y_n\|^2\big] \leq \mathbb{E}\big[\|X_0 - Y_0\|^2\big] \mathrm{e}^{\varepsilon(\vartheta, \Delta t)t_n}, \tag{9.48}$$

where

$$\varepsilon(\vartheta, \Delta t) = \frac{1}{\Delta t} \ln \gamma(\vartheta, \Delta t)$$

and

$$\gamma(\vartheta, \Delta t) = \beta(\vartheta, \Delta t) + \frac{3\widetilde{M}\Delta t^2}{4(1 - 2\vartheta\mu_f \Delta t)},$$

being $\widetilde{M} = \max\{\widetilde{M}_1, \widetilde{M}_2\}$, with $\widetilde{M}_1$ defined as

$$\widetilde{M}_1 = m \cdot \max_{j=1,\ldots,m} \sup_{[0,T]} \frac{\mathbb{E}\big[\|\Delta L^j g_n^j\|^2\big]}{\mathbb{E}\big[\|X_n - Y_n\|^2\big]} \tag{9.49}$$

and $\widetilde{M}_2$ defined as

$$\widetilde{M}_2 = m(m-1) \cdot \max_{\substack{j_1, j_2=1,\ldots,m \\ j_1 \neq j_2}} \sup_{[0,T]} \frac{\mathbb{E}\big[\|\Delta L^{j_1} g_n^{j_2}\|^2\big]}{\mathbb{E}\big[\|X_n - Y_n\|^2\big]}.$$

With analogous arguments as those provided to prove Theorem 9.10, we can demonstrate that the following results holds true [117].

Theorem 9.12 *Under the same assumptions of Theorem 9.11, for any fixed value of* $\vartheta \in [0, 1]$*, we have*

$$|\varepsilon(\vartheta, \Delta t) - \alpha| = O(\Delta t).$$

To summarize what we have obtained so far, we know that the dynamics of a nonlinear SDE satisfying the assumptions (i)–(iii) of Theorem 9.8 is well described by the exponential mean-square inequality (9.42). This inequality can also be reproduced along the numerical dynamics of stochastic ϑ-Maruyama and ϑ-Milstein methods, as proved in Theorems 9.9 and 9.11.

According to Definition 9.10, if the parameter α in (9.42) is negative, then the problem generates mean-square contractive solutions. Clearly, transferring this property also to the numerical solutions computed by ϑ-Maruyama and ϑ-Milstein methods is equivalent to respectively impose $\nu(\vartheta, \Delta t) < 0$ in (9.43) and $\varepsilon(\vartheta, \Delta t) < 0$ in (9.48). Fulfilling these two conditions requires imposing proper stepsize restrictions, according to the following definitions.

Definition 9.11 Consider a nonlinear stochastic differential equation (9.7) satisfying assumptions (i)–(iii) given in Theorem 9.8 and let X_n and Y_n, $n \geq 0$, be two numerical solutions of (9.7) computed by the stochastic ϑ-methods (9.20) or (9.47). Then, the applied method is said to generate *mean-square contractive numerical solutions* in a region $\mathcal{R} \subseteq \mathbb{R}^+$ if, for a fixed $\vartheta \in [0, 1]$,

$$\nu(\vartheta, \Delta t) < 0, \quad \forall \Delta t \in \mathcal{R},$$

for (9.20), being $\nu(\vartheta, \Delta t)$ the parameter in (9.43), or

$$\varepsilon(\vartheta, \Delta t) < 0, \quad \forall \Delta t \in \mathcal{R},$$

for (9.47), where $\varepsilon(\vartheta, \Delta t)$ is the parameter in (9.48).

Definition 9.12 For a given value of ϑ belonging to the interval [0,1], the corresponding stochastic ϑ-method (9.20) or (9.47) is *unconditionally mean-square contractive* if $\mathcal{R} = \mathbb{R}^+$.

According to Definition 9.12, if $\mathcal{R} = \mathbb{R}^+$, the applied method is capable of reproducing the exponential mean-square contractivity for every choice of the stepsize Δt, so without any stepsize restriction. On the other hand, mean-square contractivity in a region imposes a stepsize restriction depending on the amplitude of this region. Such regions have been computed in [117] for both ϑ-Maruyama and ϑ-Milstein; we summarize here the obtained results.

- As regards ϑ-Maruyama methods (9.20), following Definition 9.11 we have that mean-square contractive numerical solutions are generated if $0 < \beta(\vartheta, \Delta t) < 1$, for any Δt in $\mathcal{R}$, i.e.,

$$\mathcal{R} = \begin{cases} \left(0, \dfrac{|\alpha|}{(1-\vartheta)^2 M_f}\right), & \vartheta < 1, \\ \mathbb{R}^+, & \vartheta = 1. \end{cases} \tag{9.50}$$

  As a consequence, the implicit Euler-Maruyama method (9.22) is the only unconditionally mean-square contractive ϑ-method (a similar property for the implicit Euler method (2.32) was true also in the deterministic case [195]);
- as regards ϑ-Milstein methods (9.47), Definition 9.11 requires $0 < \gamma(\vartheta, \Delta t) < 1$, for any $\Delta t \in \mathcal{R}$, i.e.,

$$\mathcal{R} = \begin{cases} \left(0, \dfrac{4|\alpha|}{4(1-\vartheta)^2 M_f + 3\widetilde{M}}\right), & \vartheta < 1, \\ \left(0, \dfrac{4|\alpha|}{3\widetilde{M}}\right), & \vartheta = 1, \end{cases} \tag{9.51}$$

  then no stochastic ϑ-Milstein methods are unconditionally contractive, but all subject to stepsize restrictions in retaining the mean-square contractive character along their numerical dynamics.

As visible from (9.50) and (9.51), the computation of the regions of mean-square contractivity $\mathcal{R}$ relies on the knowledge of the Lipschitz constant L_g to the diffusion of Eq. (9.7), the one-sided Lipschitz constant μ_f of its drift, the constants M and $\widetilde{M}$ defined by (9.46) and (9.49), respectively. In [117], an estimation strategy based on global optimization arguments [346] has been proposed and here briefly summarized for the estimation of the Lipschitz constant L_g; the methodologies to estimate the other constants are rather similar and the reader can find them in [117].

- **Step 1**. We perform M paths of the ϑ-methods (9.20) or (9.47) and denote by $X_n^{i,j}$ the i-th component of the j-th realization of the solution X_n, $i = 1, 2, \ldots, d$, $j = 1, 2, \ldots, M$. Then, we compute

$$a_i = \min_{j=1,\ldots,M} \min_{t_n \in \mathcal{I}_{\Delta t}} X_n^{i,j}, \qquad b_i = \max_{j=1,\ldots,M} \max_{t_n \in \mathcal{I}_{\Delta t}} X_n^{i,j}, \quad i = 1, 2, \ldots, d.$$

- **Step 2**. We generate Q couples of vectors

$$x_k = \left[x_k^1,\ x_k^2,\ \ldots,\ x_k^d\right]^\intercal, \quad y_k = \left[y_k^1,\ y_k^2,\ \ldots,\ y_k^d\right]^\intercal,$$

  with $k = 1, 2, \ldots Q$, such that (x_k^i, y_k^i) is uniformly distributed in $\left[a_i, b_i\right] \times \left[a_i, b_i\right]$, $i = 1, 2, \ldots, d$.
- **Step 3**. We compute

$$s_k = \frac{|g(x_k) - g(y_k)|^2}{|x_k - y_k|^2}, \qquad k = 1, 2, \ldots, Q.$$

- **Step 4**. We assume as estimate of L_g the value of $\max\{s_1, \ldots, s_Q\}$.

Example 9.10 Let us consider the stochastic Ginzburg-Landau model (9.10), with $\beta = -4$, $\gamma = 1$ and $\delta = 1$. It is possible to prove that this problem satisfies the assumptions (i)–(iii) of Theorem 9.8, see [38, 117, 212, 222]. In particular, the drift is one-sided Lipschitz with $\mu_f = -4$ and the diffusion is globally Lipschitz with $L_g = 1$. Then, since $\alpha = 2\mu_f + L_g = -7 < 0$, the problem generates exponentially mean-square contractive solutions, according to Definition 9.10. The constant M in (9.46) and $\widetilde{M}$ in (9.49) have also been estimated (see [117]) and their computed values are respectively equal to 16 and 1.

We first consider the stochastic trapezoidal method (9.21), i.e., the ϑ-Maruyama method (9.20) with $\vartheta = 1/2$. Its mean-square stability region (9.50) is then given by the interval $\mathcal{R} = [0, \frac{7}{4}]$, giving the stepsize restriction required to reproduce the exponential mean-square contractivity also along the numerical solution. This behavior is confirmed in Fig. 9.9, where the time-evolution of the mean-square deviation $\mathbb{E}\left[|X_n - Y_n|^2\right]$ in logarithmic scale is depicted for various values of Δt. It is visible that, the more Δt decreases, the more the numerical slope $\nu(\frac{1}{2}, \Delta t)$ in (9.43) tends to the exact slope α in (9.42). For values of $\Delta t > 7/4$, the mean-square deviation does not exponentially decay.

We finally apply the stochastic implicit Euler-Maruyama method (9.22), i.e., the stochastic ϑ-Maruyama method (9.20) with $\vartheta = 1$. As previously

(continued)

Example 9.10 (continued)
observed, this is an unconditionally mean-square contractive method, according to Definition 9.11. The graphs shown in Fig. 9.9 confirms this property of the method: we can indeed observe that also for $\Delta t = 2.3$ the mean-square deviation of X_n and Y_n decays exponentially. Clearly, the more Δt decreases, the more the numerical slope gets closer and closer to the exact one.

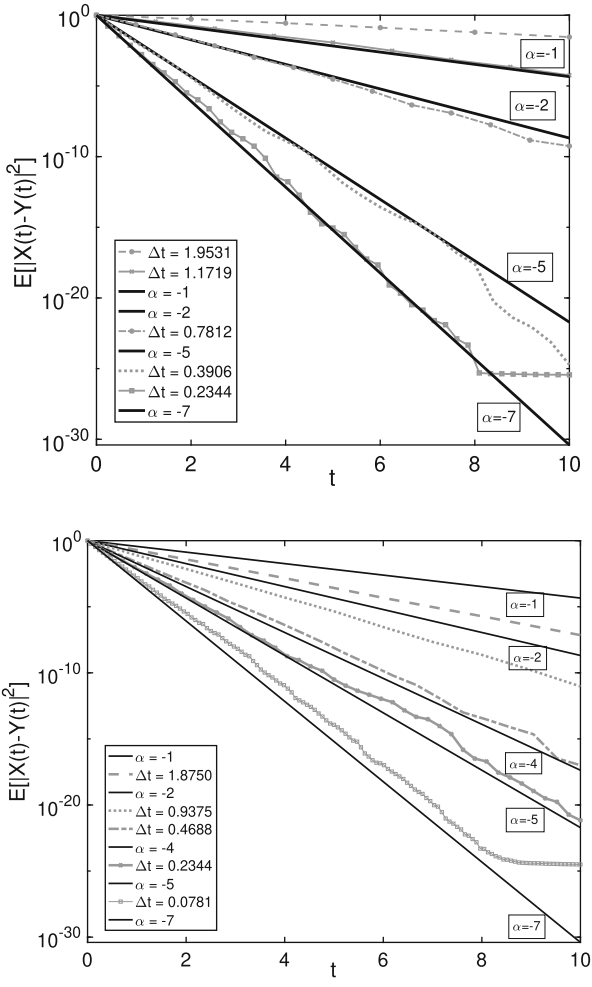


Fig. 9.9 Mean-square deviations over 2000 paths of the numerical solutions X_n and Y_n to problem (9.10) with initial values $X_0 = 1$ and $Y_0 = 0$, computed by the stochastic trapezoidal method (9.21) (top) and the implicit Euler-Maruyama method (9.22) (bottom). The y-axes is displayed in the logarithmic scale

9.7.3 Nonlinear Stability of Stochastic Runge-Kutta Methods

Let us now move to the analysis of stochastic Runge-Kutta methods applied to nonlinear SDEs (9.7), in order to investigate the mean-square contractive properties they are eventually able to maintain along the numerical dynamics, following the results provided in [118]. In the following theorem we aim to understand if the stochastic perturbation of an algebraically stable RK method (4.8) provides a good candidate to compute mean-square contractive numerical solutions. The proof is here given for a single Wiener process ($m = 1$) and the interested reader can find the general proof in [118].

Theorem 9.13 *Let us consider a nonlinear SDE (9.7) satisfying assumptions (i)–(iii) of Theorem 9.8 and two distinct initial values X_0 and Y_0, with $\mathbb{E}\left[\|X_0\|^2\right] < \infty$ and $\mathbb{E}\left[\|Y_0\|^2\right] < \infty$, leading to two solutions of (9.7), denoted as $X(t)$ and $Y(t)$, respectively. For a given SRK method (9.24), arising from the stochastic perturbation of an algebraically-stable deterministic RK method (4.8), if the matrix*

$$N = Q\Gamma + \Gamma^{\intercal} Q - qq^{\intercal},$$

with $Q = \mathrm{diag}(q)$, is symmetric positive semi-definite and the equality

$$B\Gamma + A^{\intercal} Q = bq^{\intercal}$$

is satisfied, then the approximations of $X(t)$ and $Y(t)$ computed by (9.24) satisfy the following inequality

$$\mathbb{E}\left[\|X_n - Y_n\|^2\right] \leq \mathbb{E}\left[\|X_{n-1} - Y_{n-1}\|^2\right] + \phi_n(h), \tag{9.52}$$

where

$$\phi_n(\Delta t) = 2\sum_{i=1}^{s} q_i \mathbb{E}\left[\Delta W_n \langle \widehat{X}_i^{[n]} - \widehat{Y}_i^{[n]}, g(\widehat{X}_i^{[n]}) - g(\widehat{Y}_i^{[n]})\rangle\right]. \tag{9.53}$$

Proof By introducing the auxiliary notation $Z_n = X_n - Y_n$, $\widehat{Z}_i^{[n]} = \widehat{X}_i^{[n]} - \widehat{Y}_i^{[n]}$, $\Delta f_i^{[n]} = f(\widehat{X}_i^{[n]}) - f(\widehat{Y}_i^{[n]})$ and $\Delta g_i^{[n]} = g(\widehat{X}_i^{[n]}) - g(\widehat{Y}_i^{[n]})$, SRK methods (9.24) read as follows:

$$\begin{cases} Z_n = Z_{n-1} + \Delta t \sum_{i=1}^{s} b_i \Delta f_i^{[n]} + \Delta W_n \sum_{i=1}^{s} q_i \Delta g_i^{[n]}, \\ \widehat{Z}_i^{[n]} = Z_{n-1} + \Delta t \sum_{j=1}^{s} a_{ij} \Delta f_j^{[n]} + \Delta W_n \sum_{j=1}^{s} \gamma_{ij} \Delta g_j^{[n]}, \quad i = 1, ..., s. \end{cases} \tag{9.54}$$

Passing to the norm and squaring side-by-side the first relation in (9.54), we get

$$\begin{aligned}\|Z_n\|^2 = \|Z_{n-1}\|^2 &+ \Delta t^2 \sum_{i,j=1}^{s} b_i b_j \langle \Delta f_i^{[n]}, \Delta f_j^{[n]} \rangle + \Delta W_n^2 \sum_{i,j=1}^{s} q_i q_j \langle \Delta g_i^{[n]}, \Delta g_j^{[n]} \rangle \\ &+ 2\Delta t \sum_{i=1}^{s} b_i \langle Z_{n-1}, \Delta f_i^{[n]} \rangle + 2\Delta W_n \sum_{i=1}^{s} q_i \langle Z_{n-1}, \Delta g_i^{[n]} \rangle \\ &+ 2\Delta t \Delta W_n \sum_{i,j=1}^{s} b_i q_j \langle \Delta f_i^{[n]}, \Delta g_j^{[n]} \rangle.\end{aligned}$$

We observe that

$$\begin{aligned}&\sum_{i=1}^{s} b_i \left\langle Z_{n-1}, \Delta f_i^{[n]} \right\rangle \\ &\quad = \sum_{i=1}^{s} b_i \left\langle \widehat{Z}_i^{[n]} - \Delta t \sum_{j=1}^{s} a_{ij} \Delta f_j^{[n]} - \Delta W_n \sum_{j=1}^{s} \gamma_{ij} \Delta g_j^{[n]}, \Delta f_i^{[n]} \right\rangle \\ &\quad = \sum_{i=1}^{s} b_i \left\langle \widehat{Z}_i^{[n]}, \Delta f_i^{[n]} \right\rangle - \Delta t \sum_{i,j=1}^{s} b_i a_{ij} \left\langle \Delta f_j^{[n]}, \Delta f_i^{[n]} \right\rangle \\ &\qquad - \Delta W_n \sum_{i,j=1}^{s} b_i \gamma_{ij} \left\langle \Delta g_j^{[n]}, \Delta f_i^{[n]} \right\rangle.\end{aligned}$$

Due to the hypothesis (ii) of Theorem 9.8, the algebraic stability of the underlying deterministic RK method and the assumption $\alpha < 0$, we have

$$\sum_{i=1}^{s} b_i \langle \widehat{Z}_n^{[n]}, \Delta f_i^{[n]} \rangle \leq 0$$

and, as a consequence,

$$\begin{aligned}&\sum_{i=1}^{s} b_i \langle Z_{n-1}, \Delta f_i^{[n]} \rangle \\ &\quad \leq -\Delta t \sum_{i,j=1}^{s} b_i a_{ij} \langle \Delta f_j^{[n]}, \Delta f_i^{[n]} \rangle - \Delta W_n \sum_{i,j=1}^{s} b_i \gamma_{ij} \langle \Delta g_j^{[n]}, \Delta f_i^{[n]} \rangle.\end{aligned}$$

Thus we gain

$$\|Z_n\|^2 \leq \|Z_{n-1}\|^2 - \Delta t^2 \sum_{i,j=1}^{s} m_{ij}\langle \Delta f_i^{n]}, \Delta f_j^{[n]}\rangle + \Delta W_n^2 \sum_{i,j=1}^{s} q_i q_j \langle \Delta g_i^{n]}, \Delta g_j^{[n]}\rangle$$

$$+ 2\Delta W_n \sum_{i=1}^{s} q_i \langle Z_{n-1}, \Delta g_i^{[n]}\rangle + 2\Delta t \Delta W_n \sum_{i,j=1}^{s} (b_i q_j - b_i \gamma_{ij})\langle \Delta f_i^{[n]}, \Delta g_j^{[n]}\rangle.$$

Since M is a positive semi-definite matrix, we have

$$\sum_{i,j=1}^{s} m_{ij}\langle \Delta f_i^{n]}, \Delta f_j^{[n]}\rangle \geq 0$$

and, finally,

$$\|Z_n\|^2 \leq \|Z_{n-1}\|^2 + \Delta W_n^2 \sum_{i,j=1}^{s} q_i q_j \langle \Delta g_i^{n]}, \Delta g_j^{[n]}\rangle + 2\Delta W_n \sum_{i=1}^{s} q_i \langle Z_{n-1}, \Delta g_i^{[n]}\rangle$$

$$+ 2\Delta t \Delta W_n \sum_{i,j=1}^{s} (b_i q_j - b_i \gamma_{ij})\langle \Delta f_i^{[n]}, \Delta g_j^{[n]}\rangle.$$

Let us recast the third summand of the right-hand side of last inequality using the second relation in (9.54), as follows:

$$2\Delta W_n \sum_{i=1}^{s} q_i \langle Z_{n-1}, \Delta g_i^{[n]}\rangle = 2\Delta W_n \sum_{i=1}^{s} q_i \langle Z_i^{[n]}, \Delta g_i^{[n]}\rangle$$

$$-2\Delta t \Delta W_n \sum_{i,j=1}^{s} q_i a_{ij} \langle \Delta f_j^{[n]}, \Delta g_i^{[n]}\rangle$$

$$-2\Delta W_n^2 \sum_{i,j=1}^{s} q_i \gamma_{ij} \langle \Delta g_i^{[n]}, \Delta g_j^{[n]}\rangle.$$

As a consequence,

$$\|Z_n\|^2 \leq \|Z_{n-1}\|^2 - \sum_{i,j=1}^{s} n_{ij}\langle \Delta g_i^{[n]}, \Delta g_j^{[n]}\rangle + 2\Delta W_n \sum_{i=1}^{s} q_i \langle \widehat{Z}_i^{[n]}, \Delta g_i^{[n]}\rangle$$

$$+ 2\Delta t \Delta W_n \sum_{i,j=1}^{s} (b_i q_j - b_i \gamma_{ij} - q_i a_{ij})\langle \Delta f_i^{[n]}, \Delta g_j^{[n]}\rangle.$$

According to the hypothesis, we have

$$\sum_{i,j=1}^{s} n_{ij}\langle \Delta g_i^{[n]}, \Delta g_j^{[n]}\rangle \geq 0$$

and

$$\sum_{i,j=1}^{s} (b_i q_j - b_i \gamma_{ij} - q_i a_{ij})\langle \Delta f_i^{[n]}, \Delta g_j^{[n]}\rangle = 0.$$

Therefore, we end up with

$$\|Z_n\|^2 \leq \|Z_{n-1}\|^2 + 2\Delta W_n \sum_{i=1}^{s} q_i \langle \widehat{Z}_i^{[n]}, \Delta g_i^{[n]}\rangle$$

and, passing to side-by-side expectation, it leads to the thesis. □

We first note that inequality (9.52) is referred to a single step of the numerical method, then no exponential terms are present. Moreover, according to Theorem 9.13, the mean-square contractive behavior of SRK methods depends on the magnitude of a spurious term $\phi_n(\Delta t)$ that might affect the conservative character. So, it is worth investigating this issue, by means of the following results proving that the spurious term is negligible for sufficiently small values of Δt and on the long-term, so for sufficiently large time windows. The proofs are rather technical and here omitted, but the reader can find them in [118].

Theorem 9.14 *Under the assumptions (i)–(iii) of Theorem 9.8, the spurious term (9.53) satisfies the limit*

$$\lim_{\Delta t \to 0} \max_{n=0,1,\ldots,L} \phi_n(\Delta t) = 0, \tag{9.55}$$

with reference to the grid (9.17) and, for any fixed $\Delta t > 0$,

$$\lim_{n \to \infty} \phi_n(\Delta t) = 0. \tag{9.56}$$

Example 9.11 Let us consider the stochastic Ginzburg-Landau model (9.10), with $\beta = -4$, $\gamma = 1$ and $\delta = 1$, as in Example 9.10. We now address our attention to the application of selected SRK methods (9.24) arising as stochastic perturbation of algebraically stable RK methods, namely:

- the Gaussian RK method (4.24) depending on one stage, i.e., the midpoint method;
- the two-stage Gaussian RK method with two stages (4.25).

The easy check that the corresponding SRK methods fulfills the conditions of Theorem 9.13 is left to the reader.

Figure 9.10 provides the comparison of these two methods with the Euler-Maruyama method (9.18) and the stochastic trapezoidal method (9.21). The numerical evidence confirms the numerical preservation of the mean-square contractive character characterizing the dynamics of (9.10) for SRK methods having the one-stage and two-stage Gaussian methods as underlying RK methods. The superiority of the two-stage Gaussian SRK method with respect to the other ones is visible. The long-term theoretical behavior of the function $\phi_n(\Delta t)$ in (9.53) is also visible, since this function is monotonically decreasing to 0, until it reaches a plateau, due to machine precision. Moreover, Fig. 9.11 shows that, for decreasing values of Δt, taking the one-stage Gaussian SRK method as reference, the rate of exponential decay visible in the experiments gets closer and closer to the exact one.

9.7.4 *A Glance to the Numerics for Stochastic Hamiltonian Problems*

As described in Chap. 8, we have realized that Hamiltonian problems have certainly inspired deterministic geometric numerical integration. In this section we aim to briefly provide some results regarding the geometric numerical integration of stochastic Hamiltonian problems, introduced in Example 9.4.

As visible in Eq. (9.14), in the case of Itô-Hamiltonian problems (9.11) the Hamiltonian function is not preserved along the dynamics, as it happens for deterministic Hamiltonian problems. Moreover, its expectation is not maintained as well, but it grows linearly in time, according to the trace equation (9.14). It is natural to ask if this property is automatically preserved along the numerical dynamics generated by any numerical method for SDEs. A first experimental answer was given in [48], where the authors proved that, for quartic Hamiltonians, the stochastic perturbation of symplectic RK methods does not preserve the trace law and the same happens for some energy-preserving schemes. A final theoretical negative answer

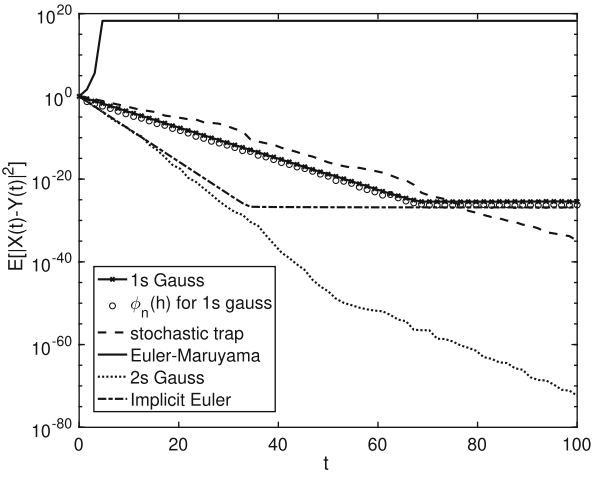


Fig. 9.10 Graphs in the semi-logarithmic scale of the mean-square deviations over 1000 paths of the numerical solutions to the stochastic Ginzburg-Landau problem (9.10) with initial values $X_0 = 1$ and $Y_0 = 0$, computed by the SRK methods (9.24) obtained as stochastic perturbation of two algebraically stable methods, i.e., Gaussian RK methods (4.24) and (4.25). These methods are compared with Euler-Maruyama method (9.18) and the stochastic trapezoidal method (9.21)

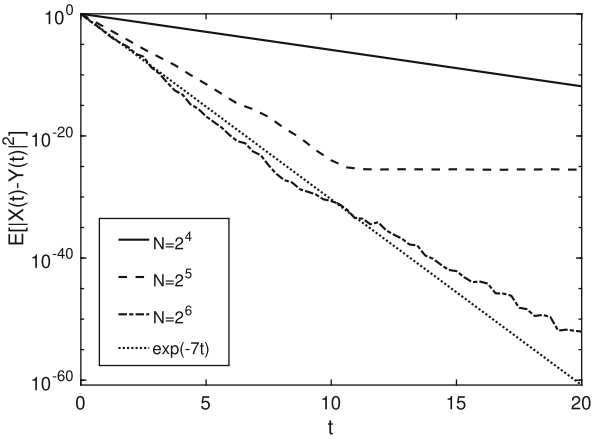


Fig. 9.11 Graphs in the semi-logarithmic scale of the mean-square deviations over 1000 paths of the numerical solutions to the stochastic Ginzburg-Landau problem (9.10) with initial values $X_0 = 1$ and $Y_0 = 0$, computed by the SRK method (9.24) obtained as stochastic perturbation of the one-stage Gaussian RK methods (4.24) for $\Delta t = 20/N$, in correspondence of different values of N. The exact rate e^{-7t} of decay of the mean-square deviation is also reported for comparison

has been provided in [136], where the authors proved what follows. For simplicity, let us consider the case of a single Wiener process

$$\begin{aligned} \mathrm{d}q(t) &= p(t)\,\mathrm{d}t \\ \mathrm{d}p(t) &= -V'(q(t))\,\mathrm{d}t + \sigma\,\mathrm{d}W(t). \end{aligned}$$

Expanding the solutions of (9.11) in power series of σ, it is possible to recognize the presence of a secular term $\sigma\sqrt{t}$ already in the linear part of the σ-expansions of p and q. Clearly, this term is more visible on long times and for large values of the stochastic term σ.

Let us see the destroying effects of the secular term in action, through the following example.

Example 9.12 (Stochastic Linear Oscillators) Let us consider a scalar damped linear stochastic oscillator, describing the motion of a particle driven by deterministic and stochastic forcing terms. The Itô SDE modelling this physical problem, given in [51, 54], has the form

$$\mathrm{d}Z(t) = QZ(t)\mathrm{d}t + \sigma q\mathrm{d}W(t), \quad t \in [0, T], \tag{9.57}$$

where

$$Z(t) = \begin{bmatrix} X(t) \\ V(t) \end{bmatrix}$$

is the vector collecting the position and velocity of the particle at time t. The matrix Q and the vector q are defined by

$$Q = \begin{bmatrix} 0 & 1 \\ -g & -\eta \end{bmatrix}, \quad q = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

being g the amplitude of the deterministic forcing term and η the value of the damping. Moreover, the parameter σ in (9.57) provides the amplitude of the stochastic forcing term, driven by the scalar Weiner process $W(t)$.

The long-term properties of (9.57), as highlighted, for instance, in [51, 54, 169], can be inferred through the analysis of the correlation matrix

$$\Gamma = \begin{bmatrix} \sigma_X^2 & \rho \\ \rho & \sigma_V^2 \end{bmatrix} = \frac{\sigma^2}{2\eta} \begin{bmatrix} g^{-1} & 0 \\ 0 & 1 \end{bmatrix}, \tag{9.58}$$

(continued)

Example 9.12 (continued)
collecting the long-term expectations

$$\sigma_X^2 = \lim_{t\to\infty} \mathbb{E}[X(t)^2], \quad \sigma_V^2 = \lim_{t\to\infty} \mathbb{E}[V(t)^2], \quad \rho = \lim_{t\to\infty} \mathbb{E}[X(t)V(t)] = 0.$$

We now aim to analyze the long-time features of the ϑ-Maruyama methods (9.20) in retaining the correlation matrix (9.58) by computing the gap with respect to the numerical correlation matrix

$$\widetilde{\Gamma}(\vartheta, \Delta t) = \begin{bmatrix} \widetilde{\sigma}_X^2 & \widetilde{\rho} \\ \widetilde{\rho} & \widetilde{\sigma}_V^2 \end{bmatrix}, \tag{9.59}$$

with

$$\widetilde{\sigma}_X^2 = \lim_{t_n\to\infty} \mathbb{E}[X_n^2], \quad \widetilde{\sigma}_V^2 = \lim_{t_n\to\infty} \mathbb{E}[V_n^2], \quad \widetilde{\rho} = \lim_{t_n\to\infty} \mathbb{E}[X_n V_n],$$

where $\{X_n\}_{n=0}^L$ and $\{V_n\}_{n=0}^L$ are the numerical solutions of (9.57) computed by (9.20), with reference to the discretized domain (9.17).

In [88], the authors have provided the following properties:

- the numerical correlation matrix (9.59) corresponding to the ϑ-Maruyama method (9.20) assumes the form

$$\widetilde{\Gamma}(\vartheta, \Delta t) = \frac{\sigma^2}{\beta g} \begin{bmatrix} g(2\vartheta-1)^2\Delta t^2 + \eta(2\vartheta-1)\Delta t + 2 & g(2\vartheta-1)\Delta t \\ g(2\vartheta-1)\Delta t & 2g \end{bmatrix},$$

  with

$$\beta = g^2(2\vartheta-1)^3\Delta t^3 + 3\eta g(2\vartheta-1)^2\Delta t^2 + 2(\eta^2+2g)(2\vartheta-1)\Delta t + 4\eta;$$

- for any value of $\vartheta \in [0, 1]$ we have that

$$\lim_{\Delta t\to 0} \|\widetilde{\Gamma}(\vartheta, \Delta t) - \Gamma\|_\infty = 0.$$

  Moreover, the stochastic trapezoidal method (9.21) exactly preserves the correlation matrix (9.58);
- for any value of $\vartheta \in [0, 1]$ we also have that

$$\lim_{\eta\to\infty} \|\widetilde{\Gamma}(\vartheta, \Delta t) - \Gamma\|_\infty = 0.$$

(continued)

Example 9.12 (continued)
However, when the stochastic term σ becomes more dominant in the right-hand side of (9.57), we have that

$$\lim_{\sigma \to \infty} \|\widetilde{\Gamma}(\vartheta, \Delta t) - \Gamma\|_\infty \neq 0.$$

In order to give an idea of the gap between Γ and $\widetilde{\Gamma}(\vartheta, \Delta t)$, let us refer to Table 9.3, reporting the value of $\|\Gamma - \widetilde{\Gamma}(\vartheta, \Delta t)\|_\infty$ for fixed values of ϑ, Δt, η, g and for varying σ. We can observe that, the more σ grows, the more the deviation between Γ and $\widetilde{\Gamma}(\vartheta, \Delta t)$ becomes larger. In other terms, if the stochastic term becomes dominant, ϑ-methods may not preserve Γ accurately, unless a small enough stepsize is chosen.

An effective tool is given by the σ-expansion to the solution of (9.57), i.e., we assume as ansatz that the exact solution can be represented as a power series of σ and, as a consequence, that a numerical solution can be seen as a truncation of this expansion up to a certain power of σ. Such a technique is quite common in deterministic numerics; we refer, for instance, to [195] and references therein.

To perform the σ-expansion, we directly act on the matrix formulation (9.57) of the problem and assume, as ansatz, that

$$Z(t) = \sum_{i \geq 0} Z_i(t) \sigma^i,$$

where the coefficients $Z_i(t)$ are vectors in $\mathbb{R}^2$. Replacing the ansatz in (9.57) leads to

$$\mathrm{d}\left(\sum_{i \geq 0} Z_i(t) \sigma^i \right) = Q \sum_{i \geq 0} Z_i(t) \sigma^i \mathrm{d}t + \sigma q \mathrm{d}W(t).$$

It is now sufficient to isolate the terms up to the linear one, obtaining the stochastic differential equations

$$\mathrm{d}Z_0(t) = Q Z_0(t) \mathrm{d}t$$

and

$$\mathrm{d}Z_1(t) = Q Z_1(t) \mathrm{d}t + \sigma q \mathrm{d}W(t),$$

in the unknowns $Z_0(t)$ and $Z_1(t)$. In particular, solving the second equation reveals the presence in $Z_1(t)$ of $\sigma\sqrt{t}$, known in the literature as *secular term*.

(continued)

Example 9.12 (continued)
Clearly, a small enough value of σ makes the secular term less dominant in the long-time; on the contrary, if the stochastic part is dominant in the right-hand side of (9.57), the secular term becomes dominant and compromises the accurate preservation of Γ, unless a really small value of Δt is chosen.

To confirm our analysis, we solve numerically (9.57) by the stochastic trapezoidal method, exactly preserving the correlation matrix (9.58). However, as visible from Table 9.4, the more σ grows, the more the method loses the excellent preservation properties achieved for more moderate values of σ. This is not surprising, according to the theoretical arguments given in [88] and briefly reported in this example. Clearly, in order to be more accurate when σ is bigger, we need to balance the presence of the secular term with a small stepsize.

We have understood that secular terms destroy the overall accuracy in numerically retaining the properties of the continuous problem under investigation. For this reason, the numerics of stochastic Hamiltonian problems deserves the design of proper solvers, able to conserve the characteristic features of the exact dynamics, such as the trace law (9.14) in the case of Itô-Hamiltonian systems (9.11).

The development of numerical methods able to preserve the trace law with respect to any Hamiltonian function has been provided, for instance, in [85], where

Table 9.3 Example 9.12: deviations between Γ and $\widetilde{\Gamma}(\vartheta, \Delta t)$ for ϑ=3/4, $\eta = g = 1$ and various values of Δt and σ

σ	$\|\Gamma - \widetilde{\Gamma}(3/4, 10^{-1})\|_\infty$	$\|\Gamma - \widetilde{\Gamma}(3/4, 10^{-2}))\|_\infty$	$\|\Gamma - \widetilde{\Gamma}(3/4, 10^{-3}))\|_\infty$
0	0	0	0
0.1	$4.73 \cdot 10^{-4}$	$4.97 \cdot 10^{-5}$	$5.00 \cdot 10^{-6}$
0.5	$1.18 \cdot 10^{-2}$	$1.24 \cdot 10^{-3}$	$1.25 \cdot 10^{-4}$
1	$4.73 \cdot 10^{-2}$	$4.97 \cdot 10^{-3}$	$5.00 \cdot 10^{-4}$
10	4.73	$6.02 \cdot 10^{+1}$	$5.00 \cdot 10^{-2}$

Table 9.4 Example: 9.12: deviations from the exact values of mean-squares positions and velocities computed by the stochastic trapezoidal method (9.21) in [0,100], with $\eta = g = 1$, $\Delta t = 100/2^{12}$ and for various values of σ. Numerical expectations have been estimated over 1000 paths

σ	$\lvert\sigma_X^2 - \widetilde{\sigma}_X^2\rvert$	$\lvert\sigma_V^2 - \widetilde{\sigma}_V^2\rvert$
10^{-6}	$1.78 \cdot 10^{-14}$	$1.83 \cdot 10^{-15}$
10^{-5}	$2.94 \cdot 10^{-12}$	$2.32 \cdot 10^{-12}$
10^{-4}	$7.00 \cdot 10^{-11}$	$4.92 \cdot 10^{-11}$
10^{-3}	$4.74 \cdot 10^{-09}$	$1.64 \cdot 10^{-08}$
10^{-2}	$1.34 \cdot 10^{-06}$	$6.08 \cdot 10^{-07}$
10^{-1}	$5.07 \cdot 10^{-05}$	$2.40 \cdot 10^{-04}$
1	$1.13 \cdot 10^{-02}$	$3.99 \cdot 10^{-02}$

the authors have introduced the following *drift-preserving integrator*

$$\begin{aligned}
\Psi_{n+1} &= p_n + \Sigma \Delta W_{n+1} - \frac{h}{2} \int_0^1 V'(q_n + sh\Psi_{n+1}) \mathrm{d}s, \\
q_{n+1} &= q_n + h\Psi_{n+1}, \\
p_{n+1} &= p_n + \Sigma \Delta W_{n+1} - h \int_0^1 V'(q_n + sh\Psi_{n+1})\, \mathrm{d}s,
\end{aligned} \tag{9.60}$$

having both strong and weak orders equal to 1, satisfying the following property, whose proof is here omitted, but the reader can find it in [85].

Theorem 9.15 *For a given Itô-Hamiltonian system (9.13), if $V \in C^1(\mathbb{R}^d)$, the drift-preserving method (9.60) satisfies the numerical trace law*

$$\mathbb{E}\left[H(p_n, q_n)\right] = \mathbb{E}\left[H(p(t_0), q(t_0))\right] + \frac{1}{2}\mathrm{Tr}\left(\Sigma^{\intercal}\Sigma\right) t_n,$$

for any grid point $t_n \in \mathcal{I}_{\Delta t}$.

Let us conclude this section by briefly considering also the case of Stratonovich Hamiltonian systems

$$\begin{cases}
\mathrm{d}q(t) = \nabla_p H(q(t), p(t)) \left[\mathrm{d}t + \Sigma \circ \mathrm{d}W(t)\right], \\
\mathrm{d}p(t) = -\nabla_q H(q(t), p(t)) \left[\mathrm{d}t + \Sigma \circ \mathrm{d}W(t)\right].
\end{cases} \tag{9.61}$$

Along the exact dynamics described by this problem, both the Hamiltonian function and its expectation are preserved [270], due to the fact the Stratonovich calculus has the same chain rule of real calculus. Long-term conservation issues along the discretization of Stratonovich Hamiltonian systems has been covered in [121], by performing a weak backward error analysis. All details are here omitted (included the construction of modified differential equations; also see [1, 140, 321, 349] and references therein), but it is worth analyzing the main result of this analysis.

Theorem 9.16 *Let us consider the Stratonowich Hamiltonian system (9.61) and let (q_n, p_n), $n = 0, 1, \ldots, L$, be any numerical approximation computed with a numerical method of weak order r, satisfying*

$$\mathbb{E}\left[H(q_n, p_n)\right] = H(q_0, p_0) + O\left(\Delta t^r\right).$$

Then, for any $n = 1, 2, \ldots, N$, the expected numerical Hamiltonian $\mathbb{E}\left[H(q_n, p_n)\right]$ satisfies the following estimate

$$\mathbb{E}\left[H(q_n, p_n)\right] = H(q_0, p_0) + O\left(\Delta t^r \mathrm{e}^{C\Delta t^r t_n}\right) + O\left(\Delta t^{r+1}\right) + O\left(C t_n \Delta t^r\right)$$

$$+ O\left(\Delta t^r t_n \, \mathrm{e}^{C\Delta t^r t_n}\right) + O\left(C\left(\Delta t^r t_n\right)^2 \mathrm{e}^{C\Delta t^r t_n}\right),$$

being C a coefficient depending on the method. Furthermore, the gap $\mathbb{E}\left[H(q_n, p_n)\right] - H(q_0, p_0)$ remains bounded on intervals of length $O\left(\Delta t^{-r}\right)$.

In other terms, according to Theorem 9.16, an exponential error growth is visible in the discretization of Stratonowich Hamiltonian system (9.61) and the error remains bounded over time windows of length $O(\Delta t^{-r})$. Let us provide an example of this property.

Example 9.13 For separable Hamiltonians

$$H(q, p) = \frac{1}{2}\sum_{i=1}^{m} p_i^2 + V(q),$$

Stratonowich Hamiltonian systems (9.61) assume the form

$$\begin{cases} \mathrm{d}q(t) = p(t)\left(\mathrm{d}t + \widetilde{\Sigma}^{\intercal} \circ \mathrm{d}W(t)\right), \\ \mathrm{d}p(t) = -\nabla_q V(q(t))\left(\mathrm{d}t + \widetilde{\Sigma}^{\intercal} \circ \mathrm{d}W(t)\right). \end{cases} \tag{9.62}$$

Let us consider the double-well potential

$$V(q) = \frac{1}{4}q^4 - \frac{1}{2}q^2 \tag{9.63}$$

(continued)

Example 9.13 (continued)
and analyze the conservation property of the following stochastic perturbation of an energy-preserving scheme able to preserve deterministic quartic Hamiltonian and introduced in the deterministic setting by E. Celledoni et al. with $r = 1$ in [81]:

$$q_{n+1} = q_n + \frac{\xi_{n+1}}{2}(p_n + p_{n+1}),$$
$$p_{n+1} = p_n - \xi_{n+1}\left(\frac{1}{6}V'(q_n) + \frac{2}{3}V'\left(\frac{q_n + q_{n+1}}{2}\right) + \frac{1}{6}V'(q_{n+1})\right), \tag{9.64}$$

where $\xi_{n+1} = \Delta t + \sigma \Delta W_{n+1}$. Figure 9.12 shows the time evolution of the absolute value of the Hamiltonian error given by $e(t_n) = H(q_n, p_n) - H(q_0, p_0)$, for selected values of Δt, confirming the exponential error growth provided in Theorem 9.16 and its boundedness over intervals of length $O(\Delta t^{-1})$. Indeed, halving the stepsize makes the interval where the error is bounded twice as bigger. Clearly, for small enough values of the stepsize, the error looks bounded for longer times.

9.8 Exercises

1. Write a software in the programming language you prefer that simulates M trajectories of the random variable

$$U(t, W(t)) = \exp\left(t^2 + \frac{1}{2}W(t)\right), \quad t \in [0, 1],$$

where $W(t)$ is a Wiener process. Compute and plot the expected value. Compare the computed expectation with the exact one [209]

$$\mathbb{E}[U(t, W(t))] = \exp\left(\frac{9}{8}t\right),$$

for increasing values of M. Comment the results.
2. Write a software in the programming language you prefer that approximates the Stratonovich integral of a given function through the quadrature formula (9.6).
3. With reference to Example 9.1, provide an experimental verification of the Martingale property (9.4) of Itô integral.

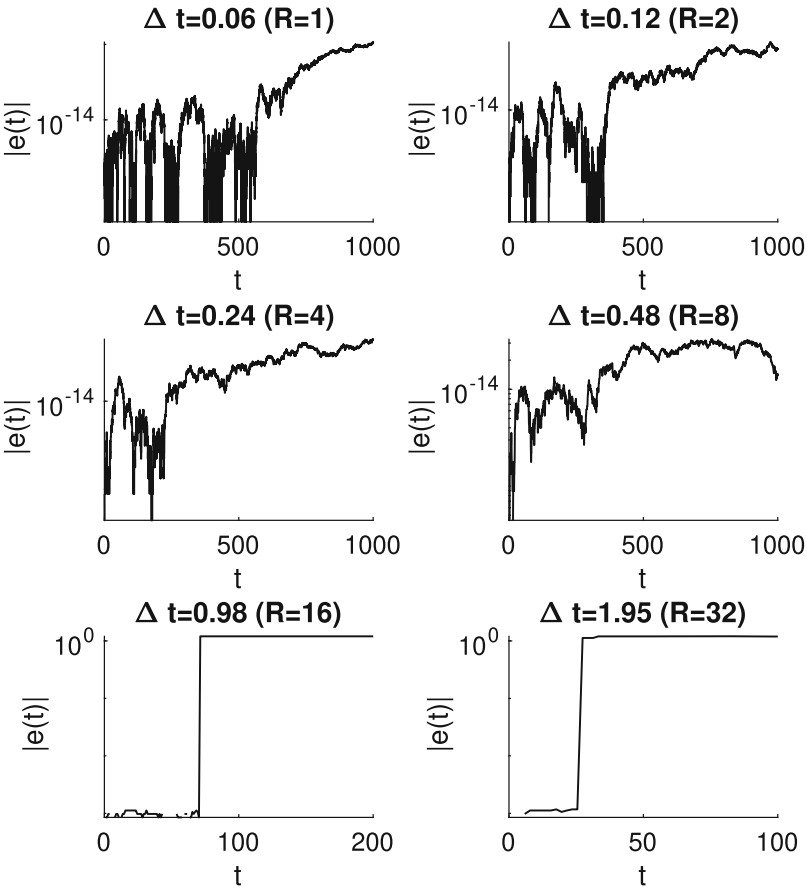


Fig. 9.12 Example 9.13: pattern in semi-logarithmic scale of the Hamiltonian deviations $|e(t)| = |H(q(t), p(t)) - H(q_0, p_0)|$ arising from the application of (9.64) to (9.62) with double-well potential problem (9.63) with $\sigma = 0.5$, $q_0 = 2$, $p_0 = 1$ and for selected values of Δt. The implementation has used $N = 2^{14}$ Wiener points and $L = N/R$ grid points, for the displayed values of R

4. Inspired by Program 9.4, write a software in the programming language you prefer that applies Milstein method (9.19) to approximate the solution of a generic SDE (9.7). Using the formulation for systems of SDEs, also extend this code to the case of SDEs. Finally, check the strong order 1 of the method by using this program.
5. Write a software in the programming language you prefer that approximates the solution of a generic SDE (9.7) by stochastic Runge-Kutta methods (9.24). Finally, fixing a specific method, estimate its strong order by using this program.

6. As explained in Sect. 9.6.3, there is a number of stochastic Runge-Kutta method (9.24) inheriting the property of A-stability (in mean-square) from the underlying deterministic Runge-Kutta method. With reference to Example 9.9, provide an experimental check of the mean-square A-stability stochastic Runge-Kutta methods arising as stochastic perturbation of the one-stage Gaussian method (4.23) and the one-stage Radau IA and IIA methods (introduced in Sect. 4.4.2).
7. Analyze the mean-square stability of ϑ-Maruyama methods (9.20), assuming that diffusion g is evaluated in X_{n+1}.
8. Analyze the mean-square contractivity properties of stochastic ϑ-Milstein methods (9.23), giving a proof of Theorem 9.11.
9. The system of SDEs (9.7) with nonlinear drift

$$f(X(t)) = -4 \begin{bmatrix} \sin(X_1(t)) \\ \\ \sin(X_2(t)) \end{bmatrix}$$

and linear diffusion

$$g(X(t)) = \frac{1}{7} \begin{bmatrix} X_1(t) & \frac{3}{2} X_2(t) \\ \\ \frac{5}{2} X_1(t) & -\frac{1}{2} X_2(t) \end{bmatrix}$$

provides exponential mean-square contractive solutions with rate $\alpha = L_g + 2\mu_f \approx -7.5$, being $L_f \approx 0.148$ and $\mu_g \approx -3.56$ (see [117]). Choosing the initial data $X_0 = [1 \;\; 1]^\intercal$ and $Y_0 = [0 \;\; 0]^\intercal$, check the preservation of mean-square contractivity along the numerical dynamics generated by the stochastic trapezoidal method (9.21) and the ϑ-Maruyama method (9.20) with $\vartheta = \frac{2}{3}$. In the computation of the mean-square stability regions (9.50), assume the value of M_f in (9.46) is equal to 16.
10. Choose a symplectic RK method (4.8), such as the two-stage Gaussian method (4.25) and construct the corresponding SRK method (9.24). Analyze its ability to preserve the trace law (9.14).

Appendix A
Summary of Test Problems

We list here all test problems selected in this book as object of all presented examples (arising from the application of the given Matlab programs) in order to analyze the performances of the presented methods and to confirm the previously given theoretical analysis. This is clearly a brief list of problems and further ones can be found in the monographs and papers cited along this book in the proper sections. These problems are collected in Matlab codes, here given for the functions `f.m`, `fp.m`, `fq.m` called in the programs presented in all previous chapters.

A.1 General ODEs

Programs 2.1, 2.2, 3.1, 4.1, 5.1 and 7.1 contain the reference to a Matlab function `f.m`, having the following structure:

```
function yp=f(problem,t,y)
switch problem
    case 'test'
        lambda=-2; % parameter to be chosen
        yp=lambda*y;
    case 'dissipative'
         A=[-5/12 125/108; -3/5 -5/12];
         yp=A*y;
    case 'prothero'
        lambda=-1e3; % parameter to be chosen
        yp=lambda*(y-sin(t))+cos(t);
    case 'pendulum'
        yp=[-sin(y(2));y(1)];
    case 'henon'
        yp=[-y(3)*(1+2*y(4));-y(3)^2+y(4)^2-y(4);y(1);
           y(2)];
    case 'vdp'
```


R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1

```
        ep=1e-3; % parameter to be chosen
        z(1)=y(2);
        z(2)=((1-y(1)^2)*y(2)-y(1))/ep;
        yp=[z(1); z(2)];
    case 'brusselator'
         A=1; % parameter to be chosen
         B=3; % parameter to be chosen
         z(1)=A+y(1)^2*y(2)-(B+1)*y(1);
         z(2)=B*y(1)-y(1)^2*y(2);
         yp=[z(1); z(2)];
end
```

A.2 Hamiltonian Problems

Program 8.1 contains the reference to two Matlab functions `fp.m` and `fq.m`, having the following structure:

```
function pdot=fp(problem,p,q)
switch problem
    case 'osc'
        omega=1; % parameter to be chosen
        pdot=-omega^2*q;
    case 'pendulum'
        pdot=-sin(q);
    case 'henon'
        pdot=[-q(1)*(1+2*q(2)); -q(1)^2+q(2)^2-q(2)];
end

function qdot=fq(problem,p,q)
switch problem
    case 'osc'
        omega=1;
        qdot=p;
    case 'pendulum'
        qdot=p;
    case 'henon'
        qdot=[p(1); p(2)];
end
```

Program 8.2 requires a supplementary `f.m` Matlab function similar to that listed above. We finally observe that programs for the geometric numerical integration of Hamiltonian problems, such as Program 8.2, require the following additional Matlab function `hamiltonian.m` with the analytic expression of the Hamiltonian.

```
function H=hamiltonian(problem,y)
switch problem
    case 'harmonicOscillator'
        omega=1;
        H=y(2)^2/2+(1/2)*y(1)^2*omega^2;
    case 'pendulum'
        H=y(1)^2/2-cos(y(2));
```

```
    case 'henonHeiles'
        H=0.5*(y(1)^2+y(2)^2+y(3)^2+y(4)^2)+y(3)^2*y
          (4)-(y(4)^3)/3;
end
```

A.3 Stochastic Differential Equations

Programs 9.4 and 9.5 contain the reference to two Matlab functions `f.m` and `g.m`, respectively containing the analytical expression of the drift and the diffusion of the problem. Their coding is analogous to the aforementioned `f.m` Matlab function for deterministic ODEs. For instance, for the geometric Brownian motion, we have

```
function drift=f(problem,y)
switch problem
    case 'geometric'
        mu=2; % parameter to be chosen
        pdot=mu*y;
end

function diffusion=g(problem,p,q)
switch problem
    case 'geometric'
        sigma=1; % parameter to be chosen
        pdot=sigma*y;
end
```

Bibliography

1. Abdulle, A., Cohen, D., Vilmart, G., Zygalakis, K.C.: High weak order methods for stochastic differential equations based on modified equations. SIAM J. Sci. Comput. **34**(3), A1800–A1823 (2012)
2. Acary, V., Brogliato, B.: Numerical Methods for Nonsmooth Dynamical Systems. Applications in Mechanics and Electronics. Springer, Berlin (2008)
3. Agarwal, R.: Difference Equations and Inequalities, 2nd edn. Dekker, New York (2000)
4. Albrecht, P.: Numerical treatment of ODEs: the theory of *A*-methods. Numer. Math. **47**(1), 59–87 (1985)
5. Albrecht, P.: A new theoretical approach to Runge-Kutta methods. SIAM J. Numer. Anal. **24**(2), 391–406 (1987)
6. Albrecht, P.: The Runge-Kutta theory in a nutshell. SIAM J. Numer. Anal. **33**(5), 1712–1735 (1996)
7. Alexander, R.: Diagonally implicit Runge-Kutta methods for stiff o.d.e.'s. SIAM J. Numer. Anal. **14**(6), 1006–1021 (1977)
8. Alexander, J.C., Seidman, T.I.: Sliding modes in intersecting switching surfaces, I: Blending. Houston J. Math. **24**(3), 545–569 (1998)
9. Alexander, J.C., Seidman, T.I.: Sliding modes in intersecting switching surfaces, II: Hysteresis. Houston J. Math. **25**(1), 185–211 (1999)
10. Almeidal, A.R.M., Amado, I.F., Reynolds, J., Berges, J., Lythe, G., Molina-Paris, C., Freitas, A.A.: Quorum-sensing in CD4+ T-cell homeostasis: a hypothesis and a model. Front. Imm. **3**, art. no. 125 (2012)
11. Anton, C.: Weak backward error analysis for stochastic Hamiltonian Systems. BIT Numer. Math. **59**, 613–646 (2019)
12. Anton, C., Deng, J., Wong, Y.S.: Weak symplectic schemes for stochastic Hamiltonian equations. Electron. Trans. Numer. **43**, 1–20 (2014)
13. Arnold, L.: Stochastic Differential Equations: Theory and Applications. Wiley, New York (1973)
14. Arnold, V.I.: Ordinary Differential Equations. MIT Press, Cambridge (1973)
15. Arnold, V.I.: Mathematical Methods of Classical Mechanics, 2nd edn. Springer, New York (1989)
16. Arnold, D.N.: Differential complexes and numerical stability. In: Tatsien, L. (ed.) Proceedings of the International Congress of Mathematicians, Vol. I: Plenary Lectures and Ceremonies, pp. 137–157. Higher Education Press, Beijing (2002)
17. Artemiev, S., Averina, T.: Numerical Analysis of Systems of Ordinary and Stochastic Differential Equations. VSP, Utrecht (1997)

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1

18. Ascher, U.M.: Numerical Methods for Evolutionary Differential Equations. SIAM, Philadelphia (2008)
19. Ascher, U.M., Petzold, L.R.: Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations. SIAM, Philadelphia (1998)
20. Atkinson, K.E.: Introduction to Numerical Analysis. Wiley, New York (1989)
21. Bazzani, A., Siboni, S., Turchetti, G.: Diffusion in Hamiltonian systems with a small stochastic perturbation. Physica D **76**(1–3), 8–21 (1994)
22. Bellen, A., Zennaro, M.: Numerical Methods for Delay Differential Equations. Oxford University Press, Oxford (2013)
23. Benettin, G., Giorgilli, A.: On the Hamiltonian interpolation of near to the identity symplectic mappings with application to symplectic integration algorithms. J. Stat. Phys. **74**, 1117–1143 (1994)
24. Birkhoff, G., Rota, G.C.: Ordinary Differential Equations. Wiley, New York (1978)
25. Birkhoff, G., Varga, R.S.: Discretization errors for well-set Cauchy problems: I. J. Math. Phys. **44**, 1–23 (1965)
26. Blanes, S., Casas, F.: A Concise Introduction to Geometric Numerical Integration. CRC Press, New York (2016)
27. Bochev, P.B., Scovel, C.: On quadratic invariants and symplectic structure. BIT **34**, 337–345 (1994)
28. Brauer, F., Castillo-Chavez, C.: Mathematical Models in Population Biology and Epidemiology. Springer, New York (2012)
29. Brauer, F., Kribs, C.: Dynamical Systems for Biological Modeling: An Introduction. Chapman and Hall/CRC, New York (2015)
30. Brauer, F., van de Driessche, P., Wu, J.: Mathematical Epidemiology. Springer, Berlin (2008)
31. Brugnano, L., Iavernaro, F.: Line integral methods which preserve all invariants of conservative problems. J. Comput. Appl. Math. **236**(16), 3905–3919 (2012)
32. Brugnano, L., Iavernaro, F.: Line Integral Methods for Conservative Problems. CRC Press, New York (2016)
33. Brugnano, L., Mazzia, F., Trigiante, D.: Fifty years of stiffness. In: Simos, T.E. (ed.) Recent Advances in Computational and Applied Mathematics, pp. 1–21. Springer, Berlin (2011)
34. Brugnano, L., Calvo, M., Montijano, J.I., Ràndez, L.: Energy-preserving methods for Poisson systems. J. Comput. Appl. Math. **236**(16), 3890–3904 (2012)
35. Brugnano, L., Iavernaro, F., Trigiante, D.: Energy-and quadratic invariants-preserving integrators based upon Gauss collocation formulae. SIAM J. Numer. Anal. **50**(6), 2897–2916 (2012)
36. Brugnano, L., Iavernaro, F., Trigiante, D.: Reprint of analysis of Hamiltonian Boundary Value Methods (HBVMs): a class of energy-preserving Runge-Kutta methods for the numerical solution of polynomial Hamiltonian systems. Commun. Nonlinear Sci. **21**(1–3), 34–51 (2015)
37. Buckwar, E., Sickenberger, T.: A comparative linear mean-square stability analysis of Maruyama- and Milstein-type methods. Math. Comput. Simul. **8**, 1110–1127 (2011)
38. Buckwar, E., D'Ambrosio, R.: Exponential mean-square stability properties of stochastic linear multistep methods. Adv. Comput. Math. **47**(6), 78 (2021)
39. Buckwar, E., Rössler, A., Winkler, R.: Stochastic Runge-Kutta methods for Itô SODEs with small noise. SIAM J. Sci. Comput. **32**, 1789–1808 (2010)
40. Budd, C.J., Iserles, A.: Geometric integration: numerical solution of differential equations on manifolds. Phil. Trans. R. Soc. A **357**, 943–1133 (1999)
41. Budd, C.J., Pigott, M.D.: Geometric integration and its applications. In: Handbook of Numerical Analysis, vol. XI, pp. 35–139. North-Holland, Amsterdam (2003)
42. Bunkin, F.V., Kadomtsev, B.B., Klimontovich, Yu.L., Koroteev, N.I., Landa, P.S., Maslov, V.P., Romanovskii, Yu.M.: In memory of Ruslan Leont'evich Stratonovich. Phys.-Usp. **40**(7), 751–752 (1997)
43. Burrage, K.: High order algebraically stable Runge-Kutta methods. SIAM J. Numer. Anal. **24**, 106–115 (1987)

44. Burrage, K.: Order properties of implicit multivalue methods for ordinary differential equations. IMA J. Numer. Anal. **8**, 43–69 (1988)
45. Burrage, K.: Parallel and Sequential Methods for Ordinary Differential Equations. Oxford Science Publications, Clarendon Press, Oxford (1995)
46. Burrage, K., Burrage, P.M.: Order conditions of stochastic Runge-Kutta methods by B-series. SIAM J. Numer. Anal. **38**, 1626–1646 (2001)
47. Burrage, K., Burrage, P.M.: Low-rank Runge-Kutta methods, symplecticity and stochastic Hamiltonian problems with additive noise. J. Comput. Appl. Math. **236**, 3920–3930 (2012)
48. Burrage, P.M., Burrage, K.: Structure-preserving Runge-Kutta methods for stochastic Hamiltonian equations with additive noise. Numer. Algorithms **65**(3), 519–532 (2014)
49. Burrage, K., Butcher, J.C.: Stability criteria for implicit Runge-Kutta methods. SIAM J. Numer. Anal. **16**(1), 46–57 (1979)
50. Burrage, K., Butcher, J.C.: Nonlinear stability of a general class of differential equation methods. BIT **20**(2), 185–203 (1980)
51. Burrage, K., Lythe, G.: Accurate stationary densities with partitioned numerical methods for stochastic differential equations. SIAM J. Numer. Anal. **47**, 1601–1618 (2009)
52. Burrage, K., Tian, T.: Implicit stochastic Runge-Kutta methods for stochastic differential equations. BIT Numer. Math. **44**, 21–39 (2004)
53. Burrage, K., Butcher, J.C., Chipman, F.H.: An implementation of singly-implicit Runge-Kutta methods. BIT **20**, 326–340 (1980)
54. Burrage, K., Lenane, I., Lythe, G.: Numerical methods for second-order stochastic differential equations. SIAM J. Sci. Comput. **29**, 245–264 (2007)
55. Butcher, J.C.: Coefficients for the study of Runge-Kutta integration processes. J. Austral. Math. Soc. **3**, 185–201 (1963)
56. Butcher, J.C.: On the integration process of A. Huta. J. Austral. Math. Soc. **3**, 202–206 (1963)
57. Butcher, J.C.: Implicit Runge-Kutta processes. Math. Comput. **18**, 50–64 (1964)
58. Butcher, J.C.: Integration processes based on Radau quadrature formulas. Math. Comput. **18**, 233–244 (1964)
59. Butcher, J.C.: On the attainable order of Runge-Kutta methods. Math. Comput. **19**, 408–417 (1965)
60. Butcher, J.C.: An algebraic theory of integration methods. Math. Comput. **26**, 79–106 (1972)
61. Butcher, J.C.: A stability property of implicit Runge-Kutta methods. BIT **15**(4), 358–361 (1975)
62. Butcher, J.C.: The Numerical Analysis of Ordinary Differential Equations. Runge-Kutta and General Linear Methods. Wiley, Chichester, NY (1987)
63. Butcher, J.C.: Linear and nonlinear stability for general linear methods. BIT **27**(2), 182–189 (1987)
64. Butcher, J.C.: General linear methods. Acta Numer. **15**, 57–256 (2006)
65. Butcher, J.C.: Thirty years of G-stability. BIT **46**, 479–489 (2006)
66. Butcher, J.C.: Forty-five years of A-stability. J. Numer. Anal. Ind. Appl. Math **4**(1–2), 1–9 (2009)
67. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 3rd edn. Wiley, Chichester (2016)
68. Butcher, J.C.: B-Series: Algebraic Analysis of Numerical Methods. Springer, Berlin (2021)
69. Butcher, J.C., D'Ambrosio, R.: Partitioned general linear methods for separable Hamiltonian problems. Appl. Numer. Math. **117**, 69–86 (2017)
70. Butcher, J.C., Gulshad, I.: Order conditions for G-symplectic methods. BIT **55**(4), 927–948 (2015)
71. Butcher, J.C., Hewitt, L.L.: The existence of symplectic general linear methods. Numer. Algorithms **51**, 77–84 (2009)
72. Butcher, J.C., Wanner, G.: Runge-Kutta methods: some historical notes. Appl. Numer. Math. **22**(1–3), 113–151 (1996)
73. Butcher, B.C., Habib, Y., Hill, A.T., Norton, T.J.T.: The control of parasitism in G-symplectic methods. SIAM J. Numer. Anal. **52**(5), 2440–2465 (2014)

74. Byrne, G.D., Hindmarsh, A.C.: Stiff ODE solvers: a review of current and coming attractions. J. Comput. Phys. **70**, 1–62 (1987)
75. Calvo, M., Montijano, J.I., Randez, L.: Algorithm 968: Disode45: a Matlab Runge-Kutta solver for piecewise smooth IVPs of Filippov type. ACM Trans. Math. Soft. **43**(3), 1–14 (2016)
76. Cannon, W.: The Wisdom of the Body. Norton, New York (1932)
77. Capasso, V.: Mathematical Structures of Epidemic Systems. Springer, Berlin (1993)
78. Cardone, A., Jackiewicz, Z., Verner, J.H., Welfert, B.: Order conditions for general linear methods. J. Comput. Appl. Math. **290**, 44–64 (2015)
79. Cash, J.R.: Efficient numerical method for the solution of stiff initial-value problems and differential algebraic equations. R. Soc. Lond. Proc. Ser. A Math. Phys. Eng. Sci. **459**(2032), 797–815 (2003)
80. Cayley, A.: On the theory of the analytical forms called trees. Philos. Mag. **13**(85), 172–176 (1857)
81. Celledoni, E., McLachlan, R.I., McLaren, D.I., Owren, B., Quispel, G.R.W., Wright, W.M.: Energy-preserving Runge-Kutta methods. ESAIM-Math. Model. Num. **43**(4), 645–649 (2009)
82. Celledoni, E., McLachlan, R.I., Owren, B., Quispel, G.R.W.: Energy-preserving integrators and the structure of B-series. Found. Comput. Math. **10**(6), 673–693 (2010)
83. Celledoni, E., Owren, B., Sun, Y.: The minimal stage, energy preserving Runge-Kutta method for polynomial Hamiltonian systems is the averaged vector field method. Math Comput. **83**(288), 1689–1700 (2014)
84. Celledoni, E., Eidnes, S., Owren, B., Ringholm, T.: Energy-preserving methods on Riemannian manifolds. Math. Comput. **89**, article number 3470 (2020)
85. Chen, C., Cohen, D., D'Ambrosio, R., Lang, A.: Drift-preserving numerical integrators for stochastic Hamiltonian systems. Adv. Comput. Math. **46**(2), 27 (2020)
86. Chicone, C.: Stability theory of ordinary differential equations. In: Meyers, R. (eds.) Encyclopedia of Complexity and Systems Science. Springer, New York (2009)
87. Chipman, F.H.: A-stable Runge-Kutta processes. BIT **11**, 348–388 (1971)
88. Citro, V., D'Ambrosio, R.: Long-term analysis of stochastic theta-methods for damped stochastic oscillators. Appl. Numer. Math. **150**, 18–26 (2020)
89. Citro, V., D'Ambrosio, R., Di Giovacchino, S.: A-stability preserving perturbation of Runge-Kutta methods for stochastic differential equations. Appl. Math. Lett. **102**, article no. 106098 (2020)
90. Coddington, E.A., Levinson, N.: Theory of Ordinary Differential Equations. McGraw-Hill, New York (1955)
91. Cohen, D.: On the numerical discretisation of stochastic oscillators. Math. Comput. Simul. **82**, 1478–1495 (2012)
92. Cohen, D., Hairer, E.: Linear energy-preserving integrators for Poisson systems. BIT **51**(1), 91–101 (2011)
93. Cohen, D., Sigg, M.: Convergence analysis of trigonometric methods for stiff second-order stochastic differential equations. Numer. Math. **121**, 1–29 (2012)
94. Connes, A., Kreimer, D.: Lessons from quantum field theory: Hopf algebras and spacetime geometries. Lett. Math. Phys. **48**, 85–96 (1999)
95. Conte, D., D'Ambrosio, R., Paternoster, B.: On the stability of theta-methods for stochastic Volterra integral equations. Discr. Cont. Dyn. Sys. B **23**(7), 2695–2708 (2018)
96. Conte, D., D'Ambrosio, R., D'Arienzo, M.P., Paternoster, B.: Multivalue mixed collocation methods. Appl. Math. Comput. **409**, article number 126346 (2021)
97. Conte, D., D'Ambrosio, R., Paternoster, B.: Improved theta-methods for stochastic Volterra integral equations. Commun. Nonlin. Sci. Numer. Simul. **93**, article number 105528 (2021)
98. Cooper, G.J.: Stability of Runge-Kutta methods for trajectory problems. IMA J. Numer. Anal. **7**, 1–13 (1987)
99. Coppel, W.A.: Stability and Asymptotic Behavior of Differential Equations. D.C. Heath, Boston (1965)

100. Coppel, W.A.: Dichotomies in Stability Theory. Lecture Notes in Mathematics, vol. 629. Springer, New York (1978)
101. Cox, J.C., Ross, S.A.: The valuation of options for alternative stochastic processes. J. Financ. Econ. **3**, 145–166 (1976)
102. Cox, J.C., Ingersoll, J.E., Ross, S.A.: A theory of the term structure of interest rates. Econometrica **53**, 385–407 (1985)
103. Crouzeix, M.: Sur la B-stabilité des méthodes de Runge-Kutta. Numer. Math. **32**(1), 75–82 (1979)
104. Cryer, C.W.: On the instability of high order backward-difference multistep methods. BIT **12**(1), 17–25 (1972)
105. Cryer, C.W.: A new class of highly stable methods: A_0-stable methods. BIT **13**, 153–159 (1973)
106. Curtiss, C.F., Hirschfelder, J.O.: Integration of stiff equations. Proc. Natl. Acad. Sci. U. S. A. **38**, 235–243 (1952)
107. Dahlquist, G.: Stability and error bounds in the numerical integration of ordinary differential equations. Doctoral thesis, Almqvist & Wiksells, Uppsala (1958); Transactions of the Royal Institute of Technology, Stockholm (1959)
108. Dahlquist, G.: A special stability problem for linear multistep methods. BIT **3**, 27–43 (1963)
109. Dahlquist, G.: A numerical method for some ordinary differential equations with large Lipschitz constants. In: Morrell, A.J.H. (ed.) Proceedings of IFIP Congress. Information Processing 68, Edinburgh, vol. 1, Mathematics, Software, pp. 183–186 (1968)
110. Dahlquist, G.: Error analysis for a class of methods for stiff nonlinear initial value problems. In: Proc. Numer. Anal. Conf. (Dundee, Scotland, 1975). Lecture Notes in Mathematics, vol. 506, pp. 60–74. Springer, New York (1976)
111. Dahlquist, G.: G-stability is equivalent to A-stability. BIT **18**(4), 384–401 (1978)
112. Dahlquist, G.: On one-leg multistep methods. SIAM J. Numer. Anal. **20**(6), 1130–1138 (1983)
113. Dahlquist, G., Björk, Å.: Numerical Methods. Prentice-Hall, Englewood Cliffs (1974)
114. Dahlquist, G., Liniger, W., Nevanlinna, O.: Stability of two-step methods for variable integration steps. SIAM J. Numer. Anal. **20**(5), 1071–1085 (1983)
115. D'Ambrosio, R.: Highly stable multistage numerical methods for functional equations: theory and implementation issues. Ph.D. Thesis, University of Salerno - Arizona State University (2010)
116. D'Ambrosio, R.: Book review: "B-Series: Algebraic Analysis of Numerical Methods" by John C. Butcher. Eur. Math. Soc. Mag. **124**, 63–64 (2022)
117. D'Ambrosio, R., Di Giovacchino, S.: Mean-square contractivity of stochastic ϑ-methods. Commun. Nonlinear Sci Numer. Simul. **96**, article number 105671 (2021)
118. D'Ambrosio, R., Di Giovacchino, S.: Nonlinear stability issues of stochastic Runge-Kutta methods. Commun. Nonlinear Sci Numer. Simul. **94**, article number 105549 (2021)
119. D'Ambrosio, R., Di Giovacchino, S.: Optimal θ-methods for mean-square dissipative stochastic differential equations. In: Gervasi, O., et al. (eds.) ICCSA 2021. Lecture Notes in Computer Science, vol. 12949, pp. 121–134. Springer Nature Switzerland, Cham (2021)
120. D'Ambrosio, R., Di Giovacchino, S.: Numerical preservation issues in stochastic dynamical systems by θ-methods. J. Comput. Dyn. **9**(2), 123–131 (2022)
121. D'Ambrosio, R., Di Giovacchino, S.: Long-term analysis of Hamiltonians under time discretizations. SIAM J. Sci. Comput. **45**(2), A257–A288 (2023)
122. D'Ambrosio, R., Hairer, E.: Long-term stability of multi-value methods for ordinary differential equations. J. Sci. Comput. **60**(3), 627–640 (2014)
123. D'Ambrosio, R., Jackiewicz, Z.: Continuous two-step Runge-Kutta methods for ordinary differential equations. Numer. Algorithms **54**(2), 169–193 (2010)
124. D'Ambrosio, R., Jackiewicz, Z.: Construction and implementation of highly stable two-step continuous methods for stiff differential systems. Math. Comput. Simul. **81**(9), 1707–1728 (2011)

125. D'Ambrosio, R., Paternoster, B.: Two-step modified collocation methods with structured coefficients matrix for Ordinary Differential Equations. Appl. Numer. Math. **62**(10), 1325–1334 (2012)
126. D'Ambrosio, R., Paternoster, B.: Multivalue collocation methods free from order reduction. J. Comput. Appl. Math. **387**, article number 112515 (2021)
127. D'Ambrosio, R., Scalone, C.: Long-term analysis of stochastic theta-methods for damped stochastic oscillators. Appl. Numer. Math. **150**, 18–26 (2020)
128. D'Ambrosio, R., Scalone, C.: Two-step Runge-Kutta methods for stochastic differential equations. Appl. Math. Comput. **403**, article no. 125930 (2021)
129. D'Ambrosio, R., Scalone, C.: On the numerical structure preservation of nonlinear damped stochastic oscillators. Numer. Algorithms **86**(3), 933–952 (2021)
130. D'Ambrosio, R., Scalone, C.: Filon quadrature for stochastic oscillators driven by time-varying forces. Appl. Numer. Math. **169**, 21–31 (2021)
131. D'Ambrosio, R., Ferro, M., Jackiewicz, Z., Paternoster, B.: Two step almost collocations methods for Ordinary Differential Equations. Numer. Algorithms **53**(2–3), 195–217 (2010)
132. D'Ambrosio, R., Esposito, E., Paternoster, B.: General linear methods for $y'' = f(y(t))$. Numer. Algorithms **61**(2), 331–349 (2012)
133. D'Ambrosio, R., Hairer, E., Zbinden, C.J.: G-symplecticity implies conjugate-symplecticity of the underlying one-step method. BIT Numer. Math. **53**, 867–872 (2013)
134. D'Ambrosio, R., De Martino, G., Paternoster, B.: Numerical integration of Hamiltonian problems by G-symplectic methods. Adv. Comput. Math. **40**(2), 553–575 (2014)
135. D'Ambrosio, R., Moccaldi, M., Paternoster, B.: Numerical preservation of long-term dynamics by stochastic two-step methods. Discr. Cont. Dyn. Sys. B **23**(7), 2763–2773 (2018)
136. D'Ambrosio, R., Giordano, G., Paternoster, B., Ventola, A.: Perturbative analysis of stochastic Hamiltonian problems under time discretizations. Appl. Math. Lett. **120**, 107223 (2021)
137. D'Ambrosio, R., Giordano, G., Mottola, S., Paternoster, B.: Stiffness analysis to predict the spread out of fake news. Future Internet **13**, 222 (2021)
138. D'Ambrosio, R., Guglielmi, N., Scalone, C.: Destabilising nonnormal stochastic differential equations. Discr. Cont. Dyn. Sys. B **28**(3), 1632–1642 (2023)
139. Daniel, J.W., Moore, R.E.: Computation and Theory in Ordinary Differential Equations. Freeman and Co., New York (1970)
140. Debussche, A., Faou, E.: Weak backward error analysis. SIAM J. Numer. Anal. **50**, 1735–1752 (2012)
141. Dekker, K., Verwer, J.G.: Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations. CWI Monographs, vol. 2. North-Holland Publishing, Amsterdam (1984)
142. de la Cruz, H., Jimenez, J.C., Zubelli, J.P.: Locally linearized methods for the simulation of stochastic oscillators driven by random forces. BIT **57**(1), 123–151 (2017)
143. Deng, J., Anton, C., Wong, Y.S.: High-order symplectic schemes for stochastic Hamiltonian systems. Commun. Comput. Phys. **16**(1), 169–200 (2014)
144. De Vogelaere, R.: Methods of integration which preserve the contact transformation property of the Hamiltonian equations, Report No. 4. Department of Mathematics, University of Notre Dame, Notre Dame, IN (1956)
145. Di Bernardo, M., Budd, C.J., Champneys, A.R., Kowalczyk, P.: Piecewise-smooth Dynamical Systems. Theory and Applications. Springer, Berlin (2008)
146. Dieci, L., Difonzo, F.: A comparison of Filippov sliding vector fields in codimension 2. J. Comput. Appl. Math. **262**, 161–179 (2014)
147. Dieci, L., Elia, C.: Periodic orbits for planar piecewise smooth dynamical systems with a line of discontinuity. J. Dyn. Differ. Equ. **26**(4), 1049–1078 (2014)
148. Dieci, L., Lopez, L.: Sliding motion in Filippov differential systems: theoretical results and a computational approach. SIAM J. Numer. Anal. **47**(3), 2023–2051 (2009)
149. Dieci, L., Lopez, L.: Sliding motion on discontinuity surfaces of high co-dimension. A general construction for selecting a Filippov vector field. Numer. Math. **117**(4), 779–811 (2011)
150. Dieci, L., Lopez, L.: A survey of numerical methods for IVPs of ODEs with discontinuous right-hand side. J. Comput. Appl. Math. **236**, 3967–3991 (2012)

151. Dieci, L., Elia, C., Lopez, L.: A Filippov sliding vector field on an attracting co-dimension 2 discontinuity surface, and a limited loss-of-attractivity analysis. J. Differ. Equ. **254**, 1800–1832 (2013)
152. Dieci, L., Elia, C., Lopez, L.: Sharp sufficient attractivity conditions for sliding on a codimension 2 discontinuity surface. Math. Comput. Simul. **110**, 3–14 (2015)
153. Dieci, L., Elia, C., Lopez, L.: Uniqueness of Filippov sliding vector field on the intersection of two surfaces in $\mathbb{R}^3$ and implications for stability of periodic orbits. J. Nonlinear Sci. **25**, 1453–1471 (2015)
154. Diekmann, O., Heesterbeek, H., Britton, T.: Mathematical Tools for Understanding Infectious Disease Dynamics. Princeton University Press, Princeton (2012)
155. Donelson, J., III, Hansen, E.: Cyclic composite multistep predictor-corrector methods. SIAM J. Numer. Anal. **8**, 137–157 (1971)
156. Ehle, B.L.: On Padè approximations to the exponential function and A-stable methods for the numerical solution of initial value problems. Research Report CSRR 2019, Dept. AACS, University of Waterloo (1969)
157. Ehle, B.L.: A-stable methods and Padè approximations to the exponential. SIAM J. Math. Anal. **4**, 671–680 (1973)
158. Eirola, T., Sanz-Serna, J.M.: Conservation of integrals and symplectic structure in the integration of differential equations by multistep methods. Numer. Math. **61**, 281–290 (1992)
159. Ekeland, K., Owren, B., Øines, E.: Stiffness detection and estimation of dominant spectrum with explicit Runge-Kutta methods. ACM Trans. Math. Softw. **24**, 368–382 (1998)
160. Elaydi, S.: An Introduction to Difference Equations, 3rd edn. Springer, New York (2005)
161. Engquist, B., Fokas, A.S., Hairer, E., Iserles, A.: Highly Oscillatory Problems. London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge (2009)
162. Enright, W.H., Jackson, K.R., Nørsett, S.P., Thomsen, P.G.: Effective solution of discontinuous IVPs using a Runge-Kutta formula pair with interpolants. Appl. Math. Comput. **27**(4), 313–335 (1988)
163. Epstein, J.M.: Nonlinear Dynamics, Mathematical Biology, and Social Science. CRC Press, Boca Raton (2018)
164. Faou, E., Hairer, E., Pham, T.: Energy conservation with non-symplectic methods: examples and counter- examples. BIT **44**(4), 699–709 (2004)
165. Filippov, A.F.: Differential Equations with Discontinuous Right-Hand Sides. Mathematics and Its Applications. Kluwer Academic, Dordrecht (1988)
166. Franceschi, J., Pareschi, L.: Spreading of fake news, competence and learning: kinetic modelling and numerical approximation. Phil. Trans. R. Soc. A. **380**, 20210159 (2022)
167. Frank, R., Schneid, J., Ueberhuber, C.W.: Order results for implicit Runge–Kutta methods applied to stiff systems. SIAM J. Numer. Anal. **22**(3), 515–534 (1985)
168. Gard, T.C.: Introduction to Stochastic Differential Equations. Marcel Dekker Inc., New York-Basel (1988)
169. Gardiner, C.W.: Handbook of Stochastic Methods, for Physics, Chemistry and the Natural Sciences, 3rd edn. Springer, Berlin (2004)
170. Gautschi, W.: Numerical Analysis, 2nd edn. Birkhäuser, Springer, New York, Dordrecht, Heidelberg, London (2012)
171. Gear, C.W.: The automatic integration of stiff ordinary differential equations. In: Morrell, A.J.H. (ed.) Information Processing 68, pp. 187–193. North-Holland, Amsterdam (1968)
172. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Englewood Cliffs (1971)
173. Gilsing, H., Shardlow, T.: SDELab: a package for solving stochastic differential equations in MATLAB. J. Comput. Appl. Math. **205**(2), 1002–1018 (2007)
174. Ginzburg, V.L., Landau, L.D.: On the theory of superconductivity. Zh. Eksperim. i Teor. Fiz. **20**, 1064–1082 (1950)
175. Gitterman, M.: The Noisy Oscillator. The First Hundred Years, From Einstein Until Now. World Scientific, Singapore (2005)
176. Glasserman, P.: Monte Carlo Methods in Financial Engineering. Springer, New York (2004)

177. Goldberg, S.: Introduction to Difference Equations. Wiley, New York (1958)
178. Gragg, W.B., Stetter, H.J.: Generalized multistep predictor-corrector methods. J. Assoc. Comput. Mach. **11**, 188–209 (1964)
179. Grindrod, P., Higham, D.J.: A dynamical systems view of network centrality. Proc. R. Soc. A **470**, 20130835 (2014)
180. Grönwall, T.H.: Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. Ann. Math. **20**(2), 292–296 (1919)
181. Guglielmi, N., Hairer, E.: Classification of hidden dynamics in discontinuous dynamical systems. SIAM J. Appl. Dyn. Syst. **14**(3), 1454–1477 (2015)
182. Guglielmi, N., Hairer, E.: Classification of hidden dynamics in discontinuous dynamical systems. SIAM J. Appl. Dyn. Syst. **14**(3), 1454–1477 (2015)
183. Guglielmi, N., Hairer, E.: Solutions leaving a codimension-2 sliding. Nonlinear Dyn. **88**(2), 1427–1439 (2017)
184. Guglielmi, N., Hairer, E.: An efficient algorithm for solving piecewise-smooth dynamical systems. Numer. Algorithms **89**, 1311–1334 (2022)
185. Guillou, A., Soulé, F.L.: La résolution numérique des problèmes differentiels aux conditions par des méthodes de collocation. RAIRO Anal. Numér. Ser. Rouge **R-3**, 17–44 (1969)
186. Gustafsson, K., Lundh, M., Söderlind, G.: A PI stepsize control for the numerical solution of ordinary differential equations. BIT **28**(2), 270–287 (1988)
187. Hairer, E.: Backward error analysis for multistep methods. Numer. Math. **84**, 199–232 (1999)
188. Hairer, E.: Symmetric linear multistep methods. BIT Numer. Math. **46**, 515–524 (2006)
189. Hairer, E.: Challenges in geometric numerical integration. Springer INdAM Series **8**, 125–135 (2014)
190. Hairer, E., Leone, P.: Order barriers for symplectic multi-value methods. In: Griffiths, D.F., Higham, D.J., Watson, G.A. (eds.) Numerical Analysis 1997, Proceedings of the 17th Dundee Biennial Conference 1997. Pitman Research Notes in Mathematics Series, vol. 380, pp. 133–149 (1998)
191. Hairer, E., Lubich, C.: Symmetric multistep methods over long times. Numer. Math. **97**, 699–723 (2004)
192. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd edn. Springer, Berlin (2006)
193. Hairer, E., Lubich, C.: Oscillations over long times in numerical Hamiltonian systems. In: Engquist, B., Fokas, A., Hairer, E., Iserles, A. (eds.) Highly Oscillatory Problems. LMS Lecture Notes Series, vol. 366, pp. 1–24. Cambridge University Press, Cambridge (2009)
194. Hairer, E., Wanner, G.: On the Butcher group and general multi-value methods. Computing **13**, 1–15 (1974)
195. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II - Stiff and Differential-Algebraic Problems. Springer, Berlin (2002)
196. Hairer, E., Wanner, G.: Geometric numerical integration illustrated by the Störmer-Verlet method. Acta Numer. **12**, 399–450 (2003)
197. Hairer, E., Zbinden, C.J.: On conjugate symplecticity of B-series integrators. IMA J. Numer. Anal. **33**(1), 57–79 (2013)
198. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I - Nonstiff Problems, 2nd edn. Springer, Berlin (1993)
199. Halanay, A.: Differential Equations. Stability, Oscillations, Time Lags. Academic, New York (1966)
200. Halanay, A., Lefschetz, S.: Differential Equations: Geometric Theory. Interscience, New York (1957)
201. Hartman, P.: Ordinary Differential Equations, 2nd edn. SIAM, Philadelphia (2002)
202. Hartung, F., Krisztin, T., Walther, H., Wu, J.: Functional differential equations with state-dependent delays: theory and applications. In: Handbook of Differential Equations: Ordinary Differential Equations, pp. 435–545. Elsevier, Amsterdam (2006)
203. Heldt, F.S., Frensing, T., Pflugmacher, A., Gröpler, R., Peschel, B., Reichl, U.: Multiscale modeling of Influenza A virus infection supports the development of direct-acting antivirals. PLOS Comput. Biol. **9**(11), e1003372 (2013)

204. Henderson, D., Plaschko, P.: Stochastic Differential Equations in Science and Engineering. World Scientific, Singapore (2006)
205. Hénon, M., Heiles, C.: Title: the applicability of the third integral of motion: some numerical experiments. Astron. J. **69**, 73–79 (1964)
206. Henrici, P.: Discrete Variable Methods in Ordinary Differential Equations. Wiley, New York (1962)
207. Higham, D.J.: Highly continuous Runge-Kutta interpolants. ACM Trans. Mat. Soft. **17**(3), 368–386 (1991)
208. Higham, D.: Mean-square and asymptotic stability of the stochastic theta method. SIAM J. Numer. Anal. **38**, 753–769 (2000)
209. Higham, D.: An algorithmic introduction to numerical simulation of stochastic differential equations. SIAM Rev. **43**(3), 525–546 (2001)
210. Higham, D.J.: An Introduction to Financial Option Valuation: Mathematics, Stochastics and Computation. Cambridge University Press, Cambridge (2004)
211. Higham, N.: Functions of Matrices. Theory and Computation. SIAM, Philadelphia (2008)
212. Higham, D.J., Kloeden, P.E.: Numerical methods for nonlinear stochastic differential equations with jumps. Numer. Math. **101**, 101–119 (2005)
213. Higham, D.J., Kloeden, P.E.: An Introduction to the Numerical Simulation of Stochastic Differential Equations. SIAM, Philadelphia (2021)
214. Higham, D.J., Trefethen, L.N.: Stiffness of ODEs. BIT **33**, 285–303 (1993)
215. Higham, D.J., Mao, X., Stuart, A.: Exponential mean-square stability of numerical solutions to stochastic differential equations. LMS J. Comput. Math. **6**, 297–313 (2013)
216. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. **117**, 500–544 (1952)
217. Holm, D.D., Tyranowski, T.M.: Stochastic discrete Hamiltonian variational integrators. BIT Numer. Math. **58**(4), 1009–1048 (2018)
218. Hong, J., Xu, D., Wang, P.: Preservation of quadratic invariants of stochastic differential equations via Runge-Kutta methods. Appl. Numer. Math. **87**, 38–52 (2015)
219. Hull, J.C.: Options, Futures, & Other Derivatives, 4th edn. Prentice Hall, Upper Saddle River (2000)
220. Hundsdorfer, W.H., Spijker, M.N.: A note on B-stability of Runge-Kutta methods. Numer. Math. **36**, 319–331 (1981)
221. Hundsdorfer, W.H., Verwer, J.: Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations. Springer, Berlin (2003)
222. Hutzenthaler, M., Jentzen, A.: Numerical approximations of SDEs with non-globally Lipschitz continuous coefficients. Mem. Am. Math. Soc. **236**, 1112 (2015)
223. Iserles, A.: A First Course in the Numerical Analysis of Differential Equations, 2nd edn. Cambridge University Press, Cambridge (2008)
224. Iserles, A., Norsett, S.P.: Order Stars. Theory and Applications. Chapman and Hall, London (1991)
225. Itô, K.: Differential equations determining a Markoff process (original Japanese: Zenkoku Sizyo Sugaku Danwakai-si). J. Pan-Japan Math. Coll. No. 1077 (1942)
226. Itô, K.: My Sixty Years in Studies of Probability Theory: Acceptance Speech of the Kyoto Prize in Basic Sciences. In: Kyoto Prizes & Inamori Grants, pp. 142–177. The Inamori Foundation (1999)
227. Ixaru, L.Gr., Vanden Berghe, G.: Exponential Fitting. Springer Netherlands, Heidelberg (2004)
228. Jackiewicz, Z.: General Linear Methods for Ordinary Differential Equations. Wiley, Hoboken, NJ (2009)
229. Jackiewicz, Z., Tracogna, S.: A general class of two-step Runge-Kutta methods for ordinary differential equations. SIAM J. Numer. Anal. **32**, 1390–1427 (1995)
230. Jeffrey, M.D.: Hidden dynamics in models of discontinuity and switching. Physica D **274**, 34–45 (2014)

231. Jeffrey, M.D.: Dynamics at a switching intersection: hierarchy, isonomy, and multiple sliding. SIAM J. Appl. Dyn. Syst. **13**(3), 1082–1105 (2014)
232. Kang, F.: On difference schemes and symplectic geometry. In: Feng, K. (ed.) Proceedings of the 1984 Beijing Symposium on Differential Geometry and Differential Equations, pp. 42–58. Science Press, Beijing (1985)
233. Kang, F., Mengzhao, Q.: Symplectic Geometric Algorithms for Hamiltonian Systems. Springer, Berlin (2010)
234. Karatzas, I., Shreve, S.E.: Brownian Motion and Stochastic Calculus, 2nd edn. Springer, New York (1991)
235. Kelley, W.G., Peterson, A.C.: Difference Equations. An Introduction with Applications, 2nd edn. Academic, London (2001)
236. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. P. R. Soc. Lond. A-Conta **115**(772), 700–721 (1927)
237. Klebaner, F.C.: Introduction to Stochastic Calculus with Applications. Imperial College Press, London (1998)
238. Klein, F.: Vergleichende Betrachtungen über neuere geometrische Forschungen. Math. Ann. **43**, 63–100 (1893)
239. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)
240. Kwok, Y.K.: Mathematical Models of Financial Derivatives, 2nd edn. Springer, Berlin (2008)
241. Lakshmikantham, V., Trigiante, D.: Theory of Difference Equations. Numerical Methods and Applications. Academic, San Diego (1988)
242. Lambert, J.D.: Numerical Methods for Ordinary Differential Systems: The Initial Value Problem. Wiley, Chichester (1991)
243. Lambert, J.D.: Computational Methods in Ordinary Differential Equations. Wiley, London (1973)
244. Lasagni, F.M.: Canonical Runge-Kutta methods. Z. Angew. Math. Phys. **39**, 952–953 (1988)
245. LaSalle, J.P.: The Stability of Dynamical Systems. SIAM, Philadelphia (1976)
246. Lawder, M.T., Ramadesigan, V., Suthar, B., Subramanian, V.R.: Extending explicit and linearly implicit ODE solvers for index-1 DAEs. Comput. Chem. Eng. **82**, 283–292 (2015)
247. Lãzaro-Camí, J.A., Ortega, J.P.: Stochastic hamiltonian dynamical systems. Rep. Math. Phys. **61**(1), 65–122 (2008)
248. Leimkuhler, B., Reich, S.: Geometric Integrators in Hamiltonian Mechanics. Cambridge University Press, Cambridge (2003)
249. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge University Press, Cambridge (2005)
250. Leone, P.: Symplecticity and symmetry of general integration methods. Ph.D. thesis. Universite de Geneve (2000)
251. LeVeque, R.J.: Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems. SIAM, Philadelphia (2007)
252. Lindelöf, E.: Sur l'application de la méthode des approximations successives aux équations différentielles ordinaires du premier ordre. C. R. Hebd. Séances Acad. Sci. **118**, 454–457 (1894)
253. Liu, Z., Moorhead, R.J., Groner, J.: An advanced evenly-spaced streamline placement algorithm. IEEE Trans. Vis. Comput. Graph. **12**(5), 965–972 (2006)
254. Lobatto, R.: Lessen over de differentiaal-en integraalrekening (Two volumes). De Gebroeders Van Cleef, Amsterdam (1851–1852)
255. Lozinskii, S.M.: Error estimates for the numerical integration of ordinary differential equations, part I. Izv. Vyss. Uceb. Zaved Matematika **6**, 52–90 (1958)
256. Ma, Q., Ding, D., Ding, X.: Symplectic conditions and stochastic generating functions of stochastic Runge-Kutta methods for stochastic Hamiltonian systems with multiplicative noise. Appl. Comput. Math. **219**, 635–643 (2012)
257. Mahmoud, H.: A model for the spreading of fake news. J. Appl. Probab. **57**(1), 332–342 (2020)

258. Mantzaris, A.V., Higham, D.J.: A model for dynamic communicators. Eur. J. Appl. Math. **23**, 659–668 (2012)
259. Mao, X.: Stochastic Differential Equations and Applications, 2nd edn. Horwood, Chichester (2007)
260. McLachlan, R.: Featured review: geometric numerical integration: structure-preserving algorithms for ordinary differential equations. SIAM Rev. **45**(4), 817–821 (2003)
261. McLachlan, R.: Perspectives on geometric numerical integration. J. Roy. Soc. New Zeal. **49**(2), 114–125 (2019)
262. McLachlan, R., Quispel, G.: Six lectures on the geometric integration of ODEs. In: Devore, R., Iserles, A., Süli, E. (eds.) Foundations of Computational Mathematics (London Mathematical Society Lecture Note Series), pp. 155–210. Cambridge University Press, Cambridge (2001)
263. McLachlan, R.I., Quispel, G.R.W.: Splitting methods. Acta Numer. **11**, 341–434 (2002)
264. McLachlan, R.I., Quispel, G.R.W.: Geometric Integrators for ODEs. J. Phys. A: Math. Gen. **39**(19), 5251–5285 (2006)
265. McLachlan, R., Modin, K., Munthe-Kaas, H., Verdier, O.: Butcher series: a story of rooted trees and numerical methods for evolution equations. Asia Pac. Math. Newsl. **7**(1), 1–11 (2017)
266. Merson, R.H.: An operational method for the study of integration processes. In: Proceedings of Conference on Data Processing and Automatic Computing Machines 1, Weapons Research Establishment, Salisbury, pp. 1–25 (1957)
267. Mikosch, T.: Elementary Stochastic Calculus (with Finance in View). World Scientific, Singapore (1998)
268. Milstein, G.N.: Numerical Integration of Stochastic Differential Equations. Kluwer Academic Publishers, Dordrecht (1995)
269. Milstein, G.N., Tretyakov, M.V.: Stochastic Numerics for Mathematical Physics. Springer, Berlin (2004)
270. Milstein, G.N., Repin, Yu.M., Tretyakov, M.V.: Numerical methods for stochastic systems preserving symplectic structure. SIAM J. Numer. Anal. **40**, 1583–1604 (2002)
271. Milstein, G.N., Repin, Yu.M., Tretyakov, M.V.: Symplectic integration of Hamiltonian systems with additive noise. SIAM J. Numer. Anal. **39**, 2066–2088 (2002)
272. Misawa, T.: Energy Conservative Stochastic Difference Scheme for Stochastic Hamiltonian Dynamical Systems. Jpn. J. Ind. Appl. Math. **17**, 119–128 (2000)
273. Misawa, T.: Symplectic integrators to stochastic Hamiltonian dynamical systems derived from composition methods. Math. Probl. Eng., article ID 384937 (2010)
274. Miyatake, Y.: An energy-preserving exponentially-fitted continuous stage Runge-Kutta method for Hamiltonian systems. BIT **54**(3), 777–799 (2014)
275. Miyatake, Y.: A derivation of energy-preserving exponentially-fitted integrators for Poisson systems. Comput. Phys. Commun. **187**, 156–161 (2015)
276. Miyatake, Y., Butcher, J.C.: A characterization of energy-preserving methods and the construction of parallel integrators for Hamiltonian systems. SIAM J. Numer. Anal. **54**(3), 1993–2013 (2016)
277. Moser, J.: Lectures on Hamiltonian systems. Mem. Am. Math. Soc. **81**, 1–60 (1968)
278. Murayama, T., Wakamiya, S., Aramaki, E., Kobayashi, R.: Modeling the spread of fake news on Twitter. PLoS ONE **16**(4), e0250419 (2021)
279. Noble, D., Varghese, A., Kohl, P., Noble, P.: Improved guinea-pig ventricular cell model incorporating a diadic space, I_{Kr} and I_{Ks}, and length- and tension-dependent processes. Can. J. Cardiol. **14**, 123–134 (1998)
280. Øksendal, B.: Stochastic Differential Equations. An Introduction with Applications, 6th edn. Springer, Berlin (2003)
281. Oliver, J.: A curiosity of low order explicit Runge-Kutta methods. Math. Comput. **29**, 1032–1036 (1975)
282. Papakostas, S.N., Tsitouras, Ch.: Highly continuous interpolants for one-step ode solvers and their application to Runge-Kutta methods. SIAM J. Numer. Anal. **34**(1), 22–47 (1997)

283. Paternoster, B.: Present state-of-the-art in exponential fitting. A contribution dedicated to Liviu Ixaru on his 70-th anniversary. Comput. Phys. Commun. **183**, 2499–2512 (2012)
284. Peano, G.: Sull'integrabilità delle equazioni differenziali del primo ordine. Atti Accad. Sci. Torino **21**, 437–445 (1886)
285. Peano, G.: Démonstration de l'intégrabilité des équations différentielles ordinaires. Math. Ann. **37**(2), 182–228 (1890)
286. Perko, L.: Differential Equations and Dynamical Systems, 2nd edn. Springer, New York (1996)
287. Petzold, L.R., Jay, L.O., Yen, J.: Numerical solution of highly oscillatory ordinary differential equations. Acta Numer. **6**, 437–483 (1997)
288. Picard, E.: Mémoire sur la théorie des équations aux dérivées partielles et la méthode des approximations successives. J. Math. Pures Appl. **6**, 145–210 (1890)
289. Platen, P: An introduction to numerical methods for stochastic differential equations. Acta Numer. **8**, 197–246 (1999)
290. Platen, E., Bruti-Liberati, N: Numerical Solution of Stochastic Differential Equations with Jumps in Finance. Springer, Berlin (2010)
291. Prothero, A., Robinson, A.: On the stability and the accuracy of one-step methods for solving stiff systems of ordinary differential equations. Math. Comput. **28**, 145–162 (1974)
292. Quarteroni, A., Sacco, R., Saleri, F.: Numerical Mathematics, 2nd edn. Springer, New York (2007)
293. Quirynen, R., Vukov, M., Zanon, M., Diehl, M.: Autogenerating microsecond solvers for nonlinear MPC: a tutorial using ACADO integrators. Optim. Contr. Appl. Meth. **36**(5), 685–704 (2015)
294. Quispel, G.R.W., McLaren, D.I.: A new class of energy-preserving numerical integration methods. J. Phys. A-Math. Theor. **41**(4), article number 045206 (2008)
295. Radau, R.: Etude sur les formules d'approximation qui servent à calculer la valeur numérique d'une intégrale définie. J. Math. Pures Appl. **6**, 283–336 (1880)
296. Ralston, A.: Runge-Kutta Methods with Minimum Error Bounds. Math. Comput. **16**, 431–437 (1962)
297. Richtmyer, R.D., Morton, K.W.: Difference Methods for Initial-Value Problems, 2nd edn. Interscience Publishers John Wiley & Sons Inc., New York (1967)
298. Ronveaus, A.: Heun's Differential Equations. Oxford University Press, New York (1995)
299. Rössler, A.: Runge-Kutta methods for Itô stochastic differential equations with scalar noise. BIT Numer. Math. **46**, 97–110 (2006)
300. Rössler, A.: Second order Runge-Kutta methods for Ito stochastic differential equations. SIAM J. Numer. Anal. **47**, 1713–1738 (2009)
301. Rössler, A.: Runge-Kutta methods for the strong approximation of solutions of stochastic differential equations. SIAM J. Numer. Anal. **48**, 922–952 (2010)
302. Rudin, W.: Principles of Mathematical Analysis, 3rd edn. McGraw-Hill, New York (2015)
303. Rümelin, W.: Numerical treatment of stochastic differential equations. SIAM J. Numer. Anal. **19**, 604–613 (1982)
304. Runge, C.D. Tolmé: Über die numerische Auflösung von Differentialgleichungen. Math. Ann. **46**(2), 167–178 (1895)
305. Ruth, R.: A canonical integration technique. IEEE Trans. Nucl. Sci. **30**, 2669–2671 (1983)
306. Saito, Y., Mitsui, T.: Stability analysis of numerical schemes for stochastic differential equations. SIAM J. Numer. Anal. **33**, 333–344 (1996)
307. Sanz-Serna, J.M.: Runge-Kutta schemes for Hamiltonian systems. BIT **28**, 877–883 (1988)
308. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman & Hall, London (1994)
309. Särkkä, S., Solin, A.: Applied Stochastic Differential Equations. Cambridge University Press, Cambridge (2019)
310. Scalone, C.: Positivity preserving stochastic-methods for selected SDEs. Appl. Numer. Math. **172**, 351–358 (2022)

311. Schmitt, B.A., Weiner, R.: Parallel two-step W-methods with peer variables. SIAM J. Numer. Anal. **42**, 265–282 (2004)
312. Schur, J.: Über Potenzreihen die im Innern des Einheitskreises beschränkt sind. J. Reine Angew. Math. **147**, 205–232 (1916)
313. Schurz, H.: The invariance of asymptotic laws of linear stochastic systems under discretization. Z. Angew. Math. Mech. **6**, 375–382 (1999)
314. Shampine, L.F.: Evaluation of a test set for stiff ODE solvers. ACM Trans. Math. Soft. **7**, 409–420 (1981)
315. Shampine, L.F.: What is stiffness? In: Aiken, R.C. (ed.) Stiff Computation. Oxford University Press, New York (1985)
316. Shampine, L.F.: Numerical Solution of Ordinary Differential Equations. Chapman & Hall, New York (1994)
317. Shampine, L.F., Baca, L.S.: Error estimators for stiff differential equations. J. Comput. Appl. Math. **11**(2), 197–207 (1984)
318. Shampine, L.F., Gear, C.W.: A user's view of solving stiff ordinary differential equations. SIAM Rev. **21**, 1–17 (1979)
319. Shampine, L.F., Gordon, M.K.: Computer Solution of Ordinary Differential Equations. The Initial Value Problem. W.H. Freeman and Company, San Francisco (1975)
320. Shampine, L.F., Reichelt, M.W.: The MATLAB ODE suite. SIAM J. Sci. Comput. **18**, 1–22 (1997)
321. Shardlow, T.: Modified equations for stochastic differential equations. BIT Numer. Math. **46**, 111–125 (2006)
322. Shiryayev, A.N.: Some words in memory of Professor G. Maruyama. In: Watanabe, S., Prokhorov, J.V. (eds.) Probability Theory and Mathematical Statistics. Lecture Notes in Mathematics, vol. 1299. Springer, Berlin (1988)
323. Söderlind, G.: Automatic control and adaptive time-stepping. Numer. Algorithms **31**(1–4), 281–310 (2002)
324. Söderlind, G.: Digital filters in adaptive time-stepping. ACM Trans. Math. Softw. **29**(1), 1–26 (2003)
325. Söderlind, G.: The logarithmic norm. History and modern theory. BIT Numer. Math. **46**(3), 631–652 (2006)
326. Söderlind, G., Jay, L., Calvo, M.: Stiffness 1952–2012: sixty years in search of a definition. BIT **55**(2), 531–558 (2015)
327. Soroush, V., Deb, R., Sinan, A.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
328. Southern, J., Pitt-Francis, J., Whiteley, J., Stokeley, D., Kobashi, H., Nobes, R., Kadooka, Y., Gavaghan, D.: Multi-scale computational modelling in biology and physiology. Prog. Biophys. Mol. Bio. **96**, 60–89 (2008)
329. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis, 2nd edn. Springer, New York (1993)
330. Strömmen Melbö, A.H., Higham, D.J.: Numerical simulation of a linear stochastic oscillator with additive noise. Appl. Numer. Math. **51**, 89–99 (2004)
331. Suris, Y.B.: On the conservation of the symplectic structure in the numerical solution of Hamiltonian systems (in Russian). In: Filippov, S.S. (ed.) Numerical Solution of Ordinary Differential Equations, pp. 148–160. Keldysh Institute of Applied Mathematics, USSR Academy of Sciences, Moscow (1988)
332. Szegö, G.: Orthogonal Polynomials. American Mathematical Society, New York (1967)
333. Talay, D.: Stochastic Hamiltonian systems: exponential convergence to the invariant measure, and discretization by the implicit Euler scheme. Markov Processes Relat. Fields **8**, 1–36 (2002)
334. Tanaka, H.: Professor Gisiro Maruyama, in memoriam. In: Watanabe, S., Prokhorov, J.V. (eds.) Probability Theory and Mathematical Statistics. Lecture Notes in Mathematics, vol. 1299. Springer, Berlin (1988)
335. Tang, Y.F.: The simplecticity of multistep methods. Comput. Math. Appl. **25**(3), 83–90 (1993)

336. Tang, Y.F.: Formal energy of a symplectic scheme for Hamiltonian systems and its applications (I). Comput. Math. Appl. **27**, 31–39 (1994)
337. True, H., Engsig-Karup, A.P., Bigoni, D.: On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics. Math. Comput. Simul. **95**, 78–97 (2014)
338. Vazquez-Leal, H.: Generalized homotopy method for solving nonlinear differential equations. Comput. Appl. Math. **33**(1), 275–288 (2014)
339. Verlet, L.: Computer experiments on classical fluids. Phys. Rev. **159**, 98–103 (1967)
340. Verlet, L.: La malle de Newton, Bibliothèque des sciences humaines, Gallimard (1993)
341. Vilmart, G.: Weak second order multi-revolution composition methods for highly oscillatory stochastic differential equations with additive or multiplicative noise. SIAM J. Sci. Comput. **36**, 1770–1796 (2014)
342. Wanner, G., Hairer, E., Nørsett, S.P.: Order stars and stability theorems. BIT **18**, 475–489 (1978)
343. Whitney, H.: Geometric Integration Theory. Princeton Legacy Library. Princeton University Press, Princeton (1957)
344. Widlund, O.B.: A note on unconditionally stable linear multistep methods. BIT **7**(1), 65–70 (1967)
345. Wilkinson, J.H.: Error analysis of floating-point computation. Numer. Math. **2**, 319–340 (1960)
346. Wood, G., Zhang, B.: Estimation of the Lipschitz constant of a function. J. Glob. Opt. **8**, 91–103 (1996)
347. Wright, K.: Some relationships between implicit Runge-Kutta collocation and Lanczos τ methods, and their stability properties. BIT **10**, 217–227 (1970)
348. Wright, W.M.: General linear methods with inherent Runge-Kutta stability. Ph.D. thesis, The University of Auckland, Auckland (2002)
349. Zygalakis, K.C.: On the existence and the applications of modified equations for stochastic differential equations. SIAM J. Sci. Comput. **33**, 102–130 (2011)

Index

R. D'Ambrosio, *Numerical Approximation of Ordinary Differential Problems*,
La Matematica per il 3+2 148, https://doi.org/10.1007/978-3-031-31343-1