博 士 論 文 （要 約）

Geometric Numerical Integration Methods for Energy-Driven Evolution Equations

（エネルギー関数を持つ発展方程式に対する幾何学的数値計算法）

宮武 勇登

Abstract

This thesis is about geometric numerical integration methods for energy-driven evolution equations. Geometric numerical integration methods or structure-preserving numerical methods are rather specific-purpose methods in the sense that they exactly preserve or inherit geometric properties, such as symplecticity and energy-preservation, of differential equations. The main advantage of geometric numerical integration methods is that in many cases we can expect qualitatively better numerical solutions, especially over a long period of time, than with general-purpose methods. This thesis consists of two parts.

In the first part, we consider ordinary differential equations, especially Hamiltonian systems with emphasis on their energy-preservation property. It is a natural idea to consider numerical methods which exactly inherit the property. However, the study on energy-preserving methods has a shorter history than that on other geometric integration methods such as symplectic methods. The main reason is that no Runge–Kutta method is energy-preserving and thus we have to develop energy-preserving methods in another framework. The biggest contribution of the first part is to give an algebraic characterisation of so called continuous stage Runge–Kutta methods being energy-preserving. Moreover, from a practical point of view, we construct several efficient energy-preserving methods by using the characterisation.

In the second part, we consider partial differential equations. For partial differential equations, special care must be taken for space discretisation as well as time discretisation. It is of interest to extend several existing structure-preserving numerical methods, which have been developed only on uniform meshes, to nonuniform meshes. In a finite element context, we propose a general framework for constructing energy-preserving or dissipative integrators, and further extend this framework to discontinuous Galerkin methods. We also develop theory on energy-preserving/dissipative methods on moving meshes. Furthermore, we study the treatment of nonlocal equations, tanking the Hunter–Saxton equation as our working example.

Acknowledgement

First of all, I would like to express special appreciation and thanks to my supervisor Takayasu Matsuo at the University of Tokyo. I have been supervised by him since the last semester of the undergraduate course. I am very grateful to him for his constant support and for all the discussions that we have had.

My sincere thanks also go to research collaborators. The results in Chapter 4 are joint work with John Butcher. All results in Part II are joint work with Takayasu Matsuo. In addition, the results in Section 6.4 are joint work with Yoshifumi Aimoto, and those in Chapter 8 are joint work with David Cohen and Daisuke Furihata. I would also like to thank Takaharu Yaguchi, although our collaborative work is not included in this thesis.

I hope to thank all those who have helped and encouraged me in the last five years. In particular, Kazuo Murota and Masaaki Sugihara for many valuable comments, Kensuke Aishima for supporting research environment in Mathematical Informatics 3rd Laboratory, Elena Celledoni and Brynjulf Owren for their warm support in Trondheim.

I am grateful to the members of my thesis committee: Takayasu Matsuo (Chair), Kengo Nakajima, Hiromichi Nagao, Norikazu Saito and Hideyuki Suzuki. Their fruitful comments for the early version of this thesis were valuable for enhancing the thesis.

I was supported by the Research Fellowship of the Japan Society for the Promotion of Science for Young Scientists.

Finally, I am deeply grateful to my family and friends for their warm support.

Contents

Chapter 1

Introduction

1.1 Overview

The aim of this thesis is to develop geometric numerical integration methods for energy-driven evolution equations, i.e., time-dependent ordinary/partial differential equations associated with energy functions, that arise in many research fields such as physics, chemistry, biology, engineering, economics. This first section overviews geometric numerical integration methods for ordinary differential equations (Section 1.1.1) and partial differential equations (Section 1.1.2).

1.1.1 Ordinary differential equations

The study on numerical methods for ordinary differential equations (ODEs) has been a major subject for more than three centuries. For example, Newton already considered the leap frog method for solving the equations of celestial motion, and Euler suggested what is today known as the Euler method. The 19th century and first half of the 20th century are the dawning of linear multistep methods, Runge–Kutta methods and other classical numerical methods for non-stiff problems. Modern theory of numerical integration methods started in the 1950s. For example, Dahlquist introduced the concept of stability of linear multistep methods [64], and Butcher introduced the algebraic viewpoint in the study on Runge–Kutta methods [26]. After their pioneering work, these methods were reached to a certain maturity in the 1980s. For details on the history of numerical methods for ODEs, we refer to the books [32, 99, 101].

Runge–Kutta methods and linear multistep methods are rather general-purpose methods, in the sense that they can be applied to, at least formally, every first-order ODE. On the other hand, researchers in specific areas used their own numerical methods for specific equations. For example, astronomers used the Störmer–Verlet method to simulate planetary orbits[1]. The Störmer–Verlet method often produces better numerical solutions than explicit Runge–Kutta methods, in spite of its relatively low accuracy order. In the 1980s, the mechanism of the Störmer–Verlet method was realised (e.g., [72, 73]): a numerical flow of the Störmer–Verlet method, as well as the exact flow, is a symplectic map (the definition of a symplectic map will be given later in this subsection). This discovery is now recognised as the beginning of the study on geometric numerical integration methods.

Geometric numerical integration methods, which are also called structure-preserving methods, are rather specific-purpose methods in the sense that they restrict their target ODEs to certain classes. The basic concept of geometric numerical integration methods is to design numerical integrators so that they inherit some structures of the original problems. By restricting attention to a specific class of ODEs and focusing on common structures or properties, it is possible to achieve more efficient and accurate methods then general-purpose methods. Geometric numerical integration methods have a remarkable

[1]The Störmer–Verlet method was proposed independently in different contexts by Störmer [176] and Verlet [194].

characteristic: they often produce stable and qualitatively nice numerical solutions over an extremely long period of time.

These advantages of geometric numerical integration methods, especially symplectic methods, are well illustrated by numerical experiments for the Kepler problem. The Kepler problem

$$\frac{\mathrm{d}}{\mathrm{d}t}q_1 = p_1, \quad \frac{\mathrm{d}}{\mathrm{d}t}q_2 = p_2, \quad \frac{\mathrm{d}}{\mathrm{d}t}p_1 = -\frac{q_1}{(q_1^2+q_2^2)^{3/2}}, \quad \frac{\mathrm{d}}{\mathrm{d}t}p_2 = -\frac{q_2}{(q_1^2+q_2^2)^{3/2}}$$

describes the motion of two bodies. Here one of the bodies is set at the centre of our coordinate system, and (q_1, q_2) and (p_1, p_2) are the position and momentum, respectively, of another body with a suitable normalisation. We set initial values to

$$q_1(t_0) = 1-e, \quad q_2(t_0) = 0, \quad p_1(t_0) = 0, \quad p_2(t_0) = \sqrt{(1+e)/(1-e)}$$

so that the exact solution is a 2π-periodic elliptic orbit in the (q_1, q_2)-plane with the eccentricity e $(0 \le e < 1)$. Figure 1.1 plots the numerical solutions obtained by the Störmer–Verlet method and Runge's method:

- the Störmer–Verlet method

$$\begin{aligned} p_{n+1/2} &= p_n + \frac{h}{2}f(q_n), \\ q_{n+1} &= q_n + hp_{n+1/2}, \\ p_{n+1} &= p_{n+1/2} + \frac{h}{2}f(q_{n+1}), \end{aligned}$$

  for $\frac{\mathrm{d}}{\mathrm{d}t}q = p$, $\frac{\mathrm{d}}{\mathrm{d}t}p = f(q)$, where q_n and p_n denote the numerical solutions at $t = t_0 + nh$ with a stepsize h;

- Runge's method

$$y_{n+1} = y_n + \frac{h}{2}\left(f(y_n) + f\left(y_n + \frac{h}{2}f(y_n)\right)\right),$$

  for $\frac{\mathrm{d}}{\mathrm{d}t}y = f(y)$.

Both methods are explicit and second order. Runge's method produces a slightly better numerical solution than the Störmer–Verlet method for the first period (see the left two figures of Figure 1.1). However, it is observed from the right figures that the Störmer–Verlet method remains stable after a few periods, while Runge's method becomes unstable as time passes. This difference is remarkable especially when we use a large stepsize (see the right top figure). Let us evaluate the stability of the methods from a perspective of energy. For the Kepler problem, the following quantity

$$H(q,p) = \frac{p_1^2+p_2^2}{2} - \frac{1}{\sqrt{q_1^2+q_2^2}}$$

is constant along the solution. This quantity is called Hamiltonian or energy. From Figure 1.2, we observe that the errors of the energy obtained by the Euler method and Runge's method, which are both non-symplectic, grow linearly. On the other hand, the error by the Störmer–Verlet method is bounded and no drift is observed.

As another branch of geometric numerical integration methods, energy-preserving methods have also been studied recently. Why is it of importance to consider numerical methods which preserve the Hamiltonian exactly, despite the fact that symplectic methods nearly preserve the Hamiltonian without

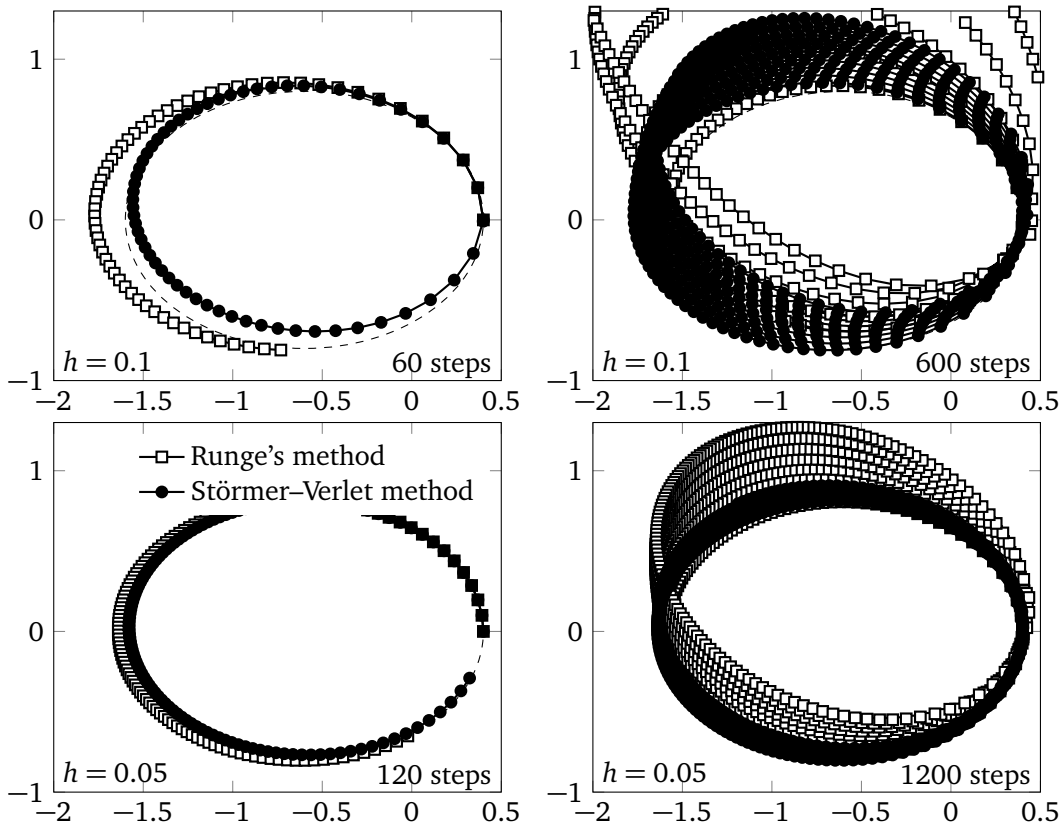


Figure 1.1: Numerical solutions on the (q_1, q_2) plane for the Kepler problem with the eccentricity $e = 0.6$ obtained by the Störmer–Verlet method and Runge's method. The exact solution is displayed by the dashed line. They are the same experiments as those in [96].

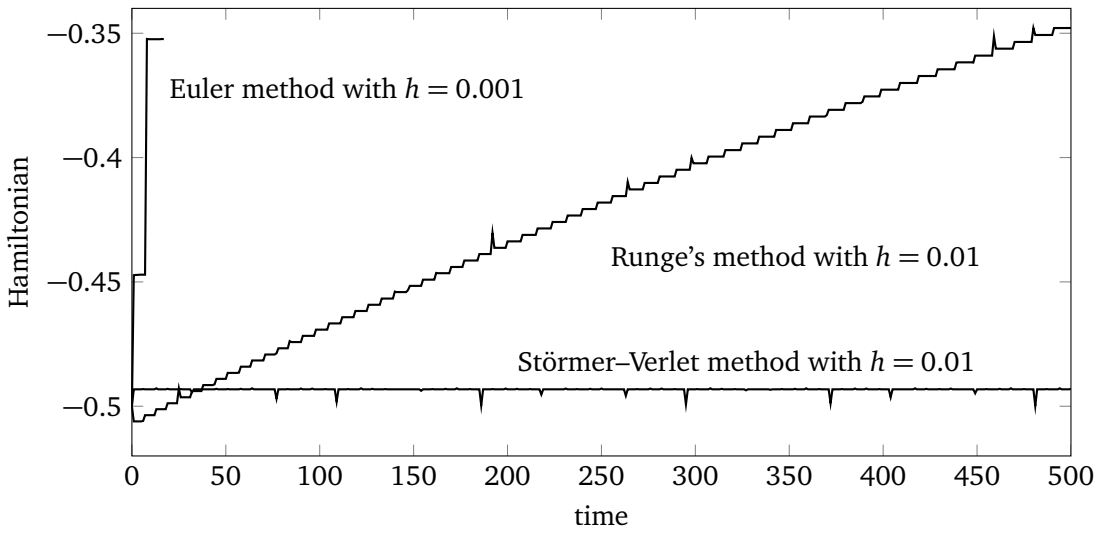


Figure 1.2: Evolutions of Hamiltonian for the Kepler problem with the eccentricity $e = 0.8$. The exact value of the Hamiltonian is 0.5 (independently of e). For the Störmer–Verlet method, several spikes are observed. In this figure, it seems that they appear randomly, but this is because the energy values are plotted every 100 steps. The spikes actually appear almost periodically when the position (q_1, q_2) gets close to the perihelion point.

drift as illustrated above?[2] There are several advantages of adopting energy-preserving methods, and below we show three of them.

- In a more general context, we sometimes need to combine the basic integration method with already-established stepsize control techniques for guaranteeing the accuracy of the numerical solutions [99, Section II.4]. It seems a natural idea to apply a standard stepsize control technique to symplectic methods. However, since a good energy-preservation of symplectic methods relies on the assumption that we use a constant stepsize, the simple combination often deteriorates the correct qualitative behaviour (see Figure 1.3). Therefore, we have to consider special strategies for changing the stepsize in order to guarantee the precise qualitative behaviour (see [97, Chapter VIII] and [123, Chapter 9]). On the other hand, energy-preserving methods preserve the Hamiltonian exactly even if the stepsize control is incorporated.

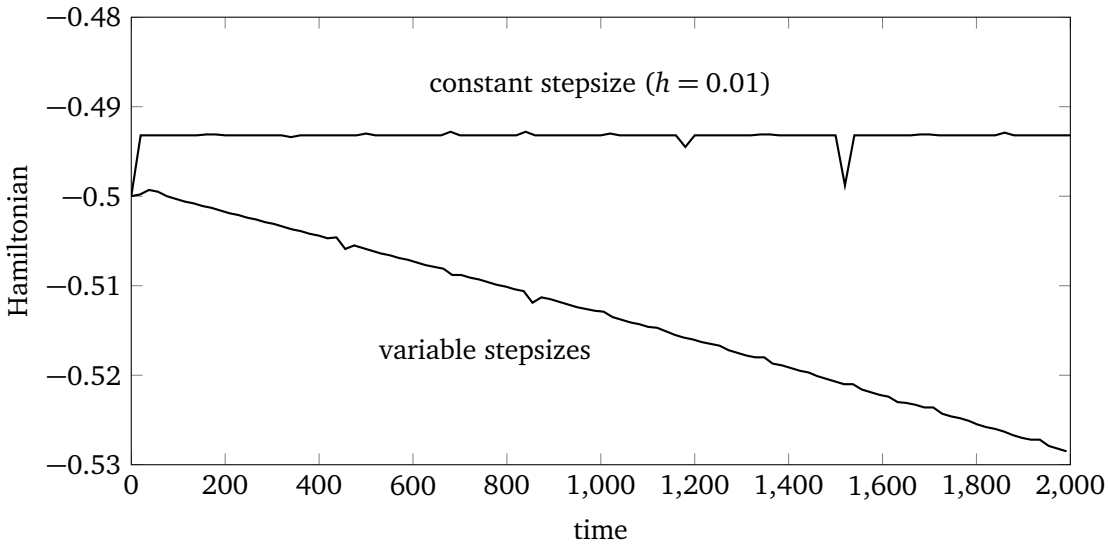


Figure 1.3: Evolutions of Hamiltonian for the Kepler problem with the eccentricity $e = 0.8$. The Störmer–Verlet method with constant stepsize ($h = 0.01$) is compared with the same method with variable stepsizes. In the latter case, the maximum of the stepsizes is smaller than 0.01 and the total number of steps is 210,872.

- Symplectic methods do not inherit the exact energy-preservation, which sometimes causes instability. We here show an example. The following experiment is taken from [167]. Let us consider the Hénon–Heiles system

$$\frac{\mathrm{d}}{\mathrm{d}t}q_1 = p_1, \quad \frac{\mathrm{d}}{\mathrm{d}t}q_2 = p_2, \quad \frac{\mathrm{d}}{\mathrm{d}t}p_1 = -q_1 - 2q_1q_2, \quad \frac{\mathrm{d}}{\mathrm{d}t}p_2 = -q_2 - q_1^2 + q_2^2,$$

  whose solution preserves the energy of the form

$$H(q,p) = \frac{p_1^2 + p_2^2}{2} + U(q), \quad U(q) = \frac{q_1^2 + q_2^2}{2} + q_1^2q_2 - \frac{1}{3}q_2^3.$$

  The Hénon–Heiles system describes a nonlinear stellar motion, and we are considering a simplified version. We set initial values to $q_1 = 0.1$, $q_2 = -0.5$, $p_1 = p_2 = 0$ so that $H = 1/6$. Figure 1.4 plots the numerical solutions obtained by the symplectic Euler method and average vector field (energy-preserving) method:

[2] It is impossible in general to construct a numerical integrator which exactly preserves both symplecticity and Hamiltonian [49, 217].

– the symplectic Euler method

$$p_{n+1} = p_n + hf(q_n),$$
$$q_{n+1} = q_n + hp_{n+1},$$

for $\frac{\mathrm{d}}{\mathrm{d}t}q = p$, $\frac{\mathrm{d}}{\mathrm{d}t}p = f(q)$;

– the average vector field method

$$y_{n+1} = y_n + h\int_0^1 f(\xi y_n + (1-\xi)y_{n+1})\,\mathrm{d}\xi,$$

for $\frac{\mathrm{d}}{\mathrm{d}t}y = f(y)$.

Since $(p_1^2 + p_2^2)/2 \geq 0$, it follows that $U(q) \leq 1/6$, which implies that the solution in the phase space is always within the thick triangle in Figure 1.4. However, since $U(q)$ sometimes exceeds $1/6$ for symplectic methods, the numerical solution might protrude from the triangle (see the left of Figure 1.4). Note that since the vertices are saddle points of U, the numerical solution immediately diverges after the jump. Such an unstable phenomenon is not observed for energy-preserving methods (see the right of Figure 1.4).

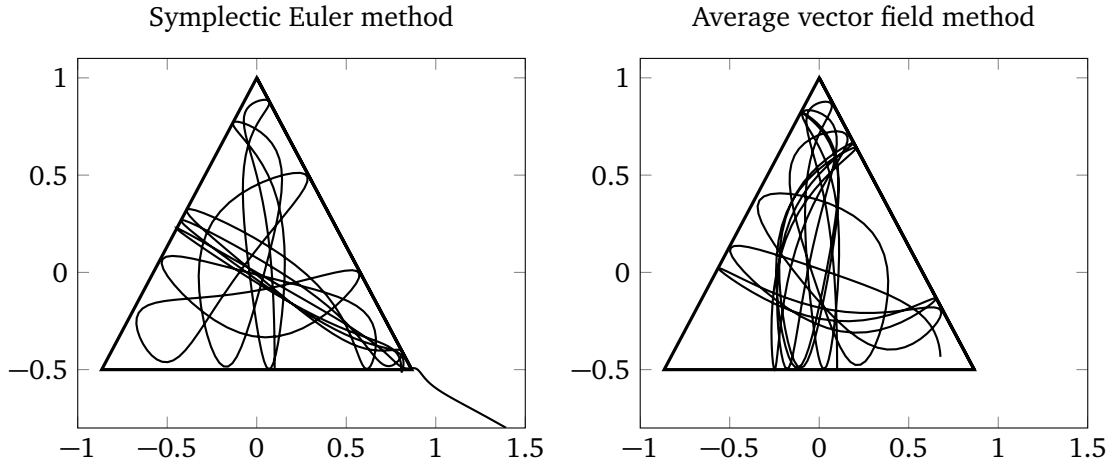


Figure 1.4: Numerical solutions in the (q_1, q_2) plane for the Hénon–Heiles system. The average vector field method exactly preserves the Hamiltonian. Initial values were set to $q_1 = 0.1$, $q_2 = -0.5$, $p_1 = p_2 = 0$ so that $H = U = 1/6$. The stepsize was $h = 0.16$ and both methods were integrated 445 times ($t_{\max} \approx 71.2$). They are the same experiments as those in [167].

- There are some ODEs whose energy is decreased along the solution. We call such a system of equations a dissipative system. For dissipative systems, we can formally apply symplectic methods, but symplectic methods cannot inherit the energy-dissipation property. On the other hand, the mechanism of energy-dissipation of dissipative systems is similar to that of the energy-preservation of energy-conservative systems. Indeed, we can construct energy-dissipative integrators for dissipative systems by using a similar idea as in energy-preserving methods for energy-conservative systems. In this sense, the study on energy-preserving methods is also useful for numerical integration of dissipative systems.

The Kepler problem and Hénon–Heiles problem are formulated as Hamiltonian systems. A Hamiltonian system is a system of ODEs of the form

$$\dot{y} = J^{-1}\nabla H(y), \qquad J = \begin{pmatrix} O & -I \\ I & 0 \end{pmatrix},$$

where $y = (q_1, \dots, q_d, p_1, \dots, p_d)^\top \in \mathbb{R}^{2d}$, the identity matrix $I \in \mathbb{R}^{d\times d}$ and zeros matrix $O \in \mathbb{R}^{d\times d}$. As briefly seen above, symplecticity and energy-preservation are two main geometric properties of Hamiltonian systems:

- symplecticity

$$\omega := \mathrm{d}q \wedge \mathrm{d}q = \sum_{i=1}^{d} \mathrm{d}q_d \wedge \mathrm{d}p_d = \text{const.},$$

- energy-preservation

$$H(q,p) = \text{const.}$$

The main target of this thesis is the Hamiltonian system. We note that many physical problems, including all equations arising in classical Newtonian mechanics, are formulated as Hamiltonian systems. Thus, numerical methods considered/developed in this thesis can be applied to a wide range of problems.

1.1.2 Partial differential equations

Geometric properties of partial differential equations (PDEs) and the history of several structure-preserving numerical methods are briefly summarised here.

Many physical problems which possess a continuous spatial structure can be described as PDEs. Since many PDEs are derived by some sort of physical principles, they often have geometric structures such as Hamiltonian structure, and possess corresponding properties such as energy-preservation/dissipation, multi-symplecticity.

As an example, let us consider the Korteweg–de Vries (KdV) equation

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(\frac{1}{2}u^2 + \frac{\partial^2 u}{\partial x^2}\right), \qquad u(t_0,\cdot) = u_0, \qquad x \in \mathbb{T},$$

which is a model of shallow water waves. The torus $\mathbb{T}$ means that we consider the periodic boundary condition. The KdV equation is energy-preserving in the sense that

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}}\left(\frac{1}{6}u^3 - \frac{1}{2}\left(\frac{\partial u}{\partial x}\right)^2\right)\mathrm{d}x = 0.$$

The KdV equation can also be written as

$$\underbrace{\begin{pmatrix} 0 & -\frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{M} z_t + \underbrace{\begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}}_{K} z_x = \nabla_z S(z),$$

where $z = (\phi, u, v, w)^\top$ and $S(z) = uw + u^3/6 + v^2/2$ (by eliminating the variables ϕ, v, w, we recover the KdV equation). Then the KdV equation is multi-symplectic in the sense that

$$\frac{\partial \omega}{\partial t} + \frac{\partial \kappa}{\partial x} = 0,$$

where $\omega = \mathrm{d}z \wedge M\mathrm{d}z$ and $\kappa = \mathrm{d}z \wedge K\mathrm{d}z$. The multi-symplecticity means that the flow of a PDE is symplectic in both time and space variables.

Similar to the ODE context, several numerical methods and schemes for PDEs proposed before 1980s are now can be regarded as structure-preserving methods. Here are some examples. The finite-difference time-domain method for the Maxwell equation proposed by Yee in 1966 [213] is now known to be symplectic. Strauss–Vazquez [177] presented an explicit energy-preserving finite difference scheme for the nonlinear Klein–Gordon equation in 1978. Delfour–Fortin–Payre [67] proposed an energy-preserving finite difference scheme for the nonlinear Schrödinger equation in 1981. Sanz-serna generalised their schemes and gave a convergence analysis in 1984 [172]. A finite element version was proposed by Akrivis–Dougalis–Karakashian in 1991 with a convergence proof [2]. Du–Nicolaides proposed an energy-dissipative finite element scheme for the Cahn–Hilliard equation in 1991 [69].

During 1990s, more general approaches for a wide class of PDEs have been introduced. Furihata [80, 81] proposed a systematic approach for constructing energy-preserving/dissipative schemes for PDEs with variational structures around 1996, which is now called the discrete variational derivative method. In 1997, Bridges introduces the concept of multi-symplecticity [18], followed by multi-symplectic discretisation methods [19].

1.2 Motivation and outline of this thesis

In this thesis, we develop geometric numerical integration methods for both ODEs and PDEs. Correspondingly, the subsequent chapters are divided into two parts. The outline of this thesis is illustrated in Figure 1.5.

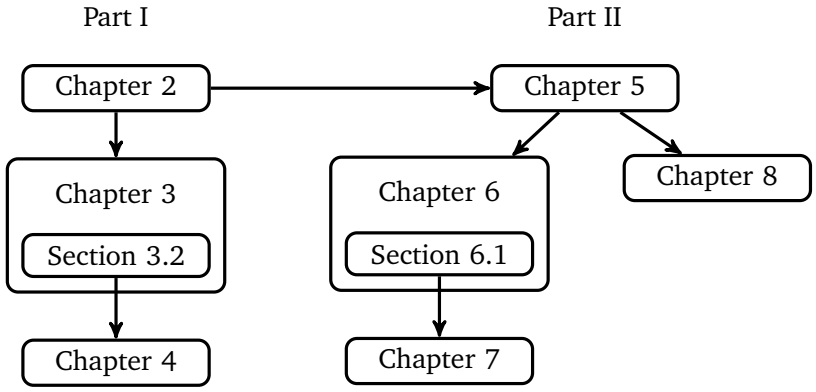


Figure 1.5: Outline of this thesis.

1.2.1 Part I: Ordinary differential equations

The main motivation of Part I is that energy-preserving methods are less developed than symplectic methods. Since energy-preserving methods have their own advantages as mentioned above, it is strongly hoped that the study on energy-preserving methods reaches to the same maturity as symplectic methods. Part I is motivated by this observation.

Chapter 2 Chapter 2 reviews basics of geometric numerical integration methods for ODEs.

Chapter 3 As we will see in Chapter 2, there is a characterisation of Runge–Kutta methods being symplectic, which has been known since 1988. However, similar characterisation for energy-preservation has not been known yet. This open problem will be solved in Chapter 3. Then by using the new characterisation we shall derive efficient integrators for ODEs whose solution exhibits periodic or oscillatory behaviour.

Chapter 4 From a different viewpoint from Chapter 3, we shall construct efficient energy-preserving methods for general Hamiltonian systems. To this end, we again use the energy-preserving characterisation proved in Chapter 3. The computational cost of the new method can further be reduced if parallelism is available.

Motivation of Chapters 3 and 4 will be explained in more detail at the end of Chapter 2.

1.2.2 Part II: Partial differential equations

Chapter 5 Chapter 5 reviews basics of geometric numerical integration methods for PDEs.

Chapter 6 Application of geometric numerical integration methods to more practical problems has been attracting attention recently. However, the existing structure-preserving methods have been developed mainly in the framework of finite difference methods, which prevented us from considering complicated domain of multidimensional problems, or using nonuniform meshes even in one-dimensional problems. Several approaches have recently been made to overcome this restriction. For example, Matsuo showed that the discrete variational derivative method mentioned above can be extended to Galerkin frameworks [135]. However, his idea had a big drawback: the target class of PDEs to which his idea is applicable is much smaller than the original discrete variational derivative method even for one-dimensional problems. In Chapter 6, we shall construct a more general framework which is free of the drawback. Furthermore, we show that the new framework can be combined with the discontinuous Galerkin methods, which allows us to derive spatially high-order energy-preserving/dissipative schemes.

Chapter 7 In Chapter 7, we shall show that several structure-preserving methods can be incorporated with grid adaptation techniques in order to make the methods more practical.

Chapter 8 In Chapter 8, we focus on a more specific equation: the Hunter–Saxton equation. Several nonlocal PDEs with rich geometric properties have been studied in some research fields. However, it is nontrivial how to apply structure-preserving numerical methods to such PDEs due to the nonlocal operators. The Hunter–Saxton equation and related equations associated with the operator ∂_x^{-2} are relatively new examples. In Chapter 8, we shall derive several structure-preserving schemes for the Hunter–Saxton and its related equations. This work is a step in advance for constructing a more general framework for nonlocal PDEs.

Motivation of Chapters 6, 7 and 8 will be explained in more detail at the end of Chapter 5.

1.2.3 Notes

Papers

New contribution of this thesis is mainly based on the author's papers. Contents are partially modified to make the thesis self-consistent. Some numerical experiments are newly done.

Chapter 3 is based on [144, 146, 145]. Chapter 4 is based on [34]. Chapter 6 is based on [1, 149]. Essential ideas of the early part of Chapter 6 were already presented in the author's master thesis [147, Chapter 4], but a more sophisticated framework is presented in this thesis. Chapter 7 is based on [152]. Chapter 8 is based on [148].

Notation

In Part I, an approximation of $y(t_0 + nh)$ with the stepsize h is denoted by y_n. In Part II, numerical solutions are denoted by $u_k^n \approx u(t_0 + n\Delta t, x_0 + k\Delta x)$ in the finite difference context, or $u^{(n)} \approx u(t_0 + n\Delta t, \cdot)$ in the finite element context. We omit the time index when considering semi-discrete schemes.

We use the following difference operators for the time derivative:

$$\delta_t^+ y_n := \frac{y_{n+1} - y_n}{h}, \qquad \delta_t^- y_n := \frac{y_n - y_{n-1}}{h}, \qquad \delta_t^{\langle 1 \rangle} y_n := \frac{y_{n+1} - y_{n-1}}{2h},$$

or

$$\delta_t^+ u_k^n := \frac{u_k^{n+1} - u_k^n}{\Delta t}, \qquad \delta_t^- u_k^n := \frac{u_k^n - u_k^{n-1}}{\Delta t}, \qquad \delta_t^{\langle 1 \rangle} u_k^n := \frac{u_k^{n+1} - u_k^{n-1}}{2\Delta t}.$$

Difference operators for the spatial derivative are defined in a similar way:

$$\delta_x^+ u_k := \frac{u_{k+1} - u_k}{\Delta x}, \qquad \delta_x^- u_k := \frac{u_k - u_{k-1}}{\Delta x}, \qquad \delta_x^{\langle 1 \rangle} u_k := \frac{u_{k+1} - u_{k-1}}{2\Delta x}.$$

Central difference operators for the high order spatial derivatives are recursively defined as

$$\delta_x^{\langle 2n+1 \rangle} u_k := \delta_x^{\langle 2n \rangle} \delta_x^{\langle 1 \rangle} u_k, \qquad \delta_x^{\langle 2n+2 \rangle} u_k := \delta_x^{\langle 2n \rangle} \delta_x^{\langle 2 \rangle} u_k,$$

with the second order central difference operator

$$\delta_x^{\langle 2 \rangle} u_k = \delta_x^+ \delta_x^- u_k = \frac{u_{k+1} - 2u_k + u_{k-1}}{\Delta x^2}.$$

The summation-by-parts formula

$$\sum_{k=0}^{N}{}'' f_k (\delta_x^+ g_k) \Delta x + \sum_{k=0}^{N}{}'' (\delta_x^- f_k) g_k \Delta x = \left[\frac{f_k (g_{k+1} + g_{k-1}) + (f_{k+1} + f_{k-1}) g^k}{4} \right]_0^N$$

is frequently used to analyse finite difference schemes for PDEs, where $\sum_{k=0}^{N}{}'' \Delta x(\cdot)$ denotes the trapezoidal rule:

$$\sum_{k=0}^{N}{}'' \Delta x f_k = \Delta x \left(\frac{1}{2} f_0 + \sum_{k=1}^{N-1} f_k + \frac{1}{2} f_N \right).$$

Part I

Geometric numerical integration methods for ODEs

Chapter 2

Preliminaries: existing methods and our motivation

In Part I, we discuss geometric numerical integration methods for ODEs. This chapter is mainly devoted to a survey of existing studies.

In Section 2.1, we introduce the fundamental idea of numerical methods for ODEs. In Section 2.2, we then briefly review basic concepts and techniques for first-order ODEs with particular emphasis on Runge–Kutta methods and B-series. Before going into geometric numerical integration methods, we give short introduction of Hamiltonian mechanics in Section 2.3. As typical examples of geometric numerical integration methods for Hamiltonian systems, we summarise symplectic methods and energy-preserving methods in Sections 2.4 and 2.5, respectively. Note that although the main interest of Part I is energy-preserving methods, it is important to study symplectic methods, and the relation between symplectic methods and energy-preserving methods for a better understanding of our motivation and the new contribution of Part I.

For other classes of geometric numerical integration methods which are not discussed in this thesis, see, e.g., [97, 123, 133].

2.1 Numerical methods for first-order ODEs

We consider a non-autonomous system of first-order differential equations

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

where $f : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N$ is a nonlinear function. Sometimes, it is convenient to use an autonomous form

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

Note that a non-autonomous form can be translated to an autonomous form by adding $\dot{t} = 1$. Since the exact solution is expressed as

$$y(t) = y_0 + \int_{t_0}^{t} f(s, y(s))\,\mathrm{d}s, \tag{2.1}$$

it seems a natural way to consider the approximation of the integral appearing in the right-hand-side when we consider numerical solutions.

We denote the time stepsize by h, and a numerical solution at $t_n = t_0 + nh$ by $y_n \approx y(t_n)$. In this thesis, we do not consider variable stepsizes. Discretising the integral of (2.1) by the rule

$$\int_a^b \phi(t)\,\mathrm{d}t \approx (b-a)\phi(a)$$

leads to the formulation

$$y_1 = y_0 + hf(t_0, y_0),$$

which is often referred to as the "Euler method" or "explicit Euler method." Other approximations give other formulae. For example, the implicit Euler method

$$y_1 = y_0 + hf(t_1, y_1)$$

is obtained by the approximation

$$\int_a^b \phi(t)\,\mathrm{d}t \approx (b-a)\phi(b).$$

The Euler methods are simple and easy to implement. However, they have various drawbacks. First, the Euler methods are first-order methods in the sense that the Taylor series of the numerical solution y_1 and exact solution $y(t_0+h)$ coincide only up to the term of h (the precise definition of order is given later in Definition 2.2). Second, the explicit Euler method is often unstable. For the harmonic oscillator, the numerical solution by the explicit Euler method diverges while the exact solution is always on the unit circle. On the other hand, although the implicit Euler method is A-stable[1], the numerical solution converges to the origin. This phenomenon illustrates that the good stability does not always imply a good qualitative behaviour.

There have been a lot of studies on the extensions of the Euler methods. Most of these studies are categorised into the following three types.

One-step methods A one-step method is a numerical integration method which is formulated as a map $y_n \mapsto y_{n+1}$. In this thesis, one-step method is often formulated as $y_0 \mapsto y_1$. A family of Runge–Kutta methods, summarised in the next section, is a typical class of one-step methods. In Part I, we mainly focus on one-step methods.

Linear multistep methods A linear multistep method is a numerical integration method which is formulated as a map $y_n, y_{n+1}, \ldots, y_{n+k} \mapsto y_{n+k+1}$. The two-step Adams–Bashdorth method

$$y_{n+2} = y_{n+1} + \frac{3}{2}f(t_{n+1}, y_{n+1}) - \frac{1}{2}f(t_n, y_n)$$

is one of the simplest examples. Reviews of the linear multistep methods are found in [32, Chapter 4] and [99, Section III].

General linear methods General linear methods are a large class of numerical methods, which contain Runge–Kutta methods and linear multistep methods as special cases. These methods were originally proposed by Gragg–Stetter [89], Butcher [27], Geer [87] and Byrne–Lambert [35] in 1964–1966. Reviews of the general linear methods are found in [31], [32, Chapter 5] and [99, Chapter III.8].

2.2 Runge–Kutta methods and B-series

Around 1900, Runge [171], Heun [103] and Kutta [121] generalised the Euler method by adding additional function evaluations in each time step. Without getting into the history of early days, we here show a general formulation.

[1] A method is said to be A-stable if it is unconditionally stable for a differential equation $\dot{y} = \lambda y$, $\mathrm{Re}\,\lambda < 0$ with a stepsize $h > 0$. A more precise definition is as follows. Let us describe a numerical method as $y_1 = R(\lambda h)y_0$. $R(z)$ is called the stability function and $S = \{z \in \mathbb{C} \mid |R(z)| \le 1\}$ is called the stability region. A method which has the property $S \supset \mathbb{C}^{-1} = \{z \mid \mathrm{Re}\,z \le 0\}$ is called A-stable. The definition of A-stability is originally due to Dahlquist [63].

Definition 2.1 (Runge–Kutta method)**.** Let b_i, a_{ij} $(i,j=1,\dots,s)$ be real numbers and $c_i=\sum_{j=1}^{s}a_{ij}$ $(i=1,\dots,s)$[2]. We search for $Y_1,\dots,Y_s$ and y_1 satisfying

$$Y_i = y_0 + h\sum_{j=1}^{s} a_{ij} f(t_0+c_j h, Y_j), \qquad i=1,\dots,s,$$
$$y_1 = y_0 + h\sum_{i=1}^{s} b_i f(t_0+c_i h, Y_i).$$

A one-step method $y_0 \mapsto y_1$ is called an s-stage Runge–Kutta method.

Note that another formulation

$$k_i = f\left(t_0+c_i h, y_0 + h\sum_{j=1}^{s} a_{ij}k_j\right), \qquad i=1,\dots,s,$$
$$y_1 = y_0 + h\sum_{i=1}^{s} b_i k_i,$$

which is mathematically equivalent to Definition 2.1, is often employed. It is customary to write the collection of a_{ij}, b_i and c_i in the Butcher tableau:

$$\begin{array}{c|c} c & A \\ \hline & b^\top \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}.$$

If the matrix A is lower triangular, one can compute the formula explicitly without solving any systems of equations. Such methods are called explicit Runge–Kutta methods. In other cases, they are referred to as implicit Runge–Kutta methods. Here are some examples of explicit Runge–Kutta methods (0 entries of A are omitted).

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array} \qquad \begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 1/2 & 1/2 \end{array} \qquad \begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & & 1/2 & & \\ 1 & & & 1 & \\ \hline & 1/6 & 2/6 & 2/6 & 1/6 \end{array}$$

Euler, order 1 Runge, order 2 Kutta, order 4

Note that the third formula is the so called Runge–Kutta method. This method is the most famous, and widely used for non-stiff problems.

Order is an important barometer of the accuracy of numerical methods.

Definition 2.2 (Order of one-step methods)**.** A Runge–Kutta method (or a general one-step method) is of order p if for a sufficiently smooth problem,

$$\|y(t_0+h)-y_1\| = \mathcal{O}(h^{p+1}) \qquad \text{as} \qquad h\to 0.$$

[2]When the coefficients depend on the stepsize h, we require $c_i=\sum_{j=1}^{s}a_{ij}+\mathcal{O}(h)$ instead.

Table 2.1: The number of order conditions.

order p	1	2	3	4	5	6	7	8	9	10
number of conditions	1	2	4	8	17	37	85	200	486	1205

This definition means that the Taylor series for the exact solution $y(t_0+h)$ coincides with that for the numerical solution y_1 up to the term h^p.

It is of interest to construct high-order Runge–Kutta formulae. At first glance, it seems that we just have to expand both the exact solution and numerical solution in the Taylor series, obtain order conditions by comparing the coefficients, and solve such conditions algebraically (i.e., not numerically). However, in general this is a tremendous task, because the number of conditions increases exponentially with the order p as shown in Table 2.1, and worse, the conditions are nonlinear.

2.2.1 B-series

B-series[3] is a powerful tool for constructing and analysing numerical methods. One of the major difficulties in analysing numerical methods is the computation of the Taylor series expansion of the numerical and exact solutions in powers of the stepsize h, because the number of the terms rapidly increases when we consider high-order terms. The main idea of the B-series is to use *rooted trees* to express such cumbersome series.

Let

$$T = \{\bullet, \ldots\}$$

be the set of rooted trees. This set is recursively defined as follows [97]:

(a) the graph $\bullet$, called a root, belongs to T;

(b) if $\tau_1, \ldots, \tau_m \in T$, then the graph obtained by connecting the roots of $\tau_1, \ldots, \tau_m$ to a new common root also belongs to T. The new tree is denoted by $\tau = [\tau_1, \ldots, \tau_m]$ (see Figure 2.1).

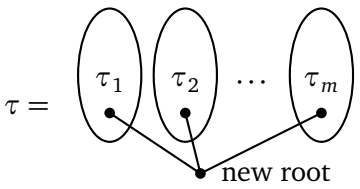


Figure 2.1: A recursively generated tree.

Note that we do not distinguish between equal (same shaped) trees. For example, we regard and mean the same tree.

We introduce the concept of symmetry coefficients and elementary differentials, and define B-series, following [97, Chapter III].

Definition 2.3 (Symmetry coefficients, e.g., [97, Chapter III])**.** The symmetry coefficient $\sigma : T \to \mathbb{R}$ is defined recursively by

$$\sigma(\bullet) = 1, \qquad \sigma(\tau) = \sigma(\tau_1)\cdots\sigma(\tau_m)\mu_1!\cdots\mu_m!,$$

where the integer μ_i denotes the number of the equal trees of τ_i.

[3] B-series was originally called Butcher-series in Hairer–Wanner [100] in honour of Butcher. The B-series theory of this section is based on [97, Chapter III]

Definition 2.4 (Elementary differentials[4], e.g., [97, Chapter III]). For a $\tau \in T$, the elementary differential is a mapping $F(\tau) : \mathbb{R}^N \to \mathbb{R}^N$, defined recursively by

$$F(\bullet)(y) = f(y),$$
$$F(\tau)(y) = f^{(m)}(F(\tau_1)(y), \ldots, F(\tau_m)(y)).$$

Using these two maps, we now define B-series.

Definition 2.5 (B-series, e.g., [97, Chapter III]). For a mapping $a : T \cup \{\emptyset\} \to \mathbb{R}$, a formal series of the form

$$B(a, y) = a(\emptyset)y + \sum_{\tau \in T} \frac{h^{|\tau|}}{\sigma(\tau)} a(\tau) F(\tau)(y) \tag{2.2}$$

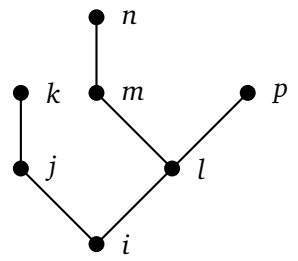
is called a B-series.

We call a discrete flow $\Phi_h(y)$, whose Taylor series is of the form (2.2) with $a(\emptyset) = 1$, a B-series integrator. A wide class of numerical methods, such as Runge–Kutta methods and the underlying one-step methods of linear multistep methods, can be interpreted as a B-series integrator. Furthermore, the exact solution can also be interpreted as a B-series integrator: the exact time-h flow of $\dot{y} = f(y)$ can be expressed as $\varphi_h(y) = B(e, y)$, where the coefficients e are given by

$$e(\emptyset) = e(\bullet) = 1, \qquad e(\tau) = \frac{1}{|\tau|} e(\tau_1) \cdots e(\tau_m) \qquad \text{for} \qquad \tau = [\tau_1, \ldots, \tau_m].$$

Every Runge–Kutta method can be interpreted as a B-series method $y_1 = B(\phi, y_0)$. The map ϕ, called the elementary weights, is represented in terms of the Runge–Kutta coefficients b and A by a simple rule. For example,

$$\phi(\quad) = \sum_{i,j,k,l,m,n,p} b_i a_{ij} a_{jk} a_{il} a_{lm} a_{mn} a_{lp}.$$



The rule is as follows. We add an index to each vertex, and multiply b_i (root) and the elements of A corresponding to all edges (e.g., a_{mn} for m-n pass). Then we sum the product with respect to all indices.

Examples of the above mappings are illustrated in Table 2.2.

The order condition of Runge–Kutta methods is summarised in terms of the elementary weights.

Theorem 2.1 (e.g., [99]). The Runge–Kutta method is of order p if and only if

$$\phi(\tau) = e(\tau) \qquad \text{for} \qquad |\tau| \le p.$$

Now we can write down the order conditions immediately based on the above theorem. However, the number of conditions is still too large to treat algebraically. Next, we consider simplifying assumptions, which are useful to check the order of implicit Runge–Kutta methods.

Let us consider $B(\rho)$, $C(\eta)$ and $D(\zeta)$ defined by

$$B(\rho): \quad \sum_{i=1}^{s} b_i c_i^{q-1} = \frac{1}{q}, \qquad q = 1, \ldots, \rho,$$
$$C(\eta): \quad \sum_{j=1}^{s} a_{ij} c_j^{q-1} = \frac{c_i^q}{q}, \qquad i = 1, \ldots, s,\ q = 1, \ldots, \eta,$$
$$D(\zeta): \quad \sum_{i=1}^{s} b_i c_i^{q-1} a_{ij} = \frac{b_j}{q}(1 - c_j^q), \qquad j = 1, \ldots, s,\ q = 1, \ldots, \zeta.$$

[4] There is a one-to-one correspondence between elementary differentials and rooted trees. This structure was first discovered by Cayley [44] in 1857, and rediscovered by Merson [142] in 1957.

Table 2.2: Trees and their related mappings.

$\lvert\tau\rvert$	τ	tree	$F(\tau)$	$\sigma(\tau)$	$e(\tau)$	$\phi(\tau)$
1	$\bullet$		f	1	1	$\sum_i b_i$
2	$[\bullet]$		$f'f$	1	1/2	$\sum_{i,j} b_i a_{ij}$
3	$[\bullet,\bullet]$		$f''(f,f)$	2	1/3	$\sum_{ijk} b_i a_{ij} a_{ik}$
3	$[[\bullet]]$		$f'f'f$	1	1/6	$\sum_{ijk} b_i a_{ij} a_{jk}$
4	$[\bullet,\bullet,\bullet]$		$f'''(f,f,f)$	6	1/4	$\sum_{ijkl} b_i a_{ij} a_{ik} a_{il}$
4	$[\bullet,[\bullet]]$		$f''(f'f,f)$	1	1/8	$\sum_{ijkl} b_i a_{ij} a_{ik} a_{kl}$
4	$[[\bullet,\bullet]]$		$f'f''(f,f)$	2	1/12	$\sum_{ijkl} b_i a_{ij} a_{jk} a_{jl}$
4	$[[[\bullet]]]$		$f'f'f'f$	1	1/24	$\sum_{ijkl} b_i a_{ij} a_{jk} a_{kl}$

Theorem 2.2 ([26])**.** The order of a Runge–Kutta method satisfying the simplifying assumption $B(\rho)$, $C(\eta)$ and $D(\zeta)$ is at least $\min(\rho, 2\eta+2, \eta+\zeta+1)$.

Symmetric methods[5] have some important properties. For example, since the order is always even, we do not have to consider the order conditions for even orders. Symmetry is defined via an adjoint method.

Definition 2.6 (Symmetric methods, e.g., [99, Chapter II.8])**.** The adjoint method Φ_h^* of Φ_h is an inverse map of the original method with reversed stepsize, i.e., $\Phi_h^* := \Phi_{-h}^{-1}$. A method satisfying $\Phi_h^* = \Phi_h$ is called symmetric.

For example, let us consider the explicit Euler method

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

We obtain the adjoint method by changing y_1 with y_0 each other and h with $-h$. This leads to the implicit Euler method

$$y_0 = y_1 - hf(y_1).$$

In this way, the adjoint method does not always coincide with the original method, and the explicit Euler method (and correspondingly the implicit Euler method) is not symmetric. The simplest symmetric method is the midpoint rule $y_1 = y_0 + hf\left(\frac{y_1+y_0}{2}\right)$.

Theorem 2.3 (e.g., [97, Chapter II, Theorem 3.2], [123, Chapter 4, Theorem 1])**.** The order of a symmetric method is always even.

Proof. Assume that Φ_h is of order p and has the following expansion

$$\Phi_h(y_0) = \varphi_h(y_0) + C(y_0)h^{p+1} + \mathcal{O}(h^{p+2}),$$

where φ_h denotes the exact flow and $C(\neq 0)$ is a smooth function. From this assumption, we can immediately evaluate the error between y_0 and $\Phi_{-h}(\varphi(y_0))$:

$$\Phi_{-h}(\varphi_h(y_0)) = y_0 + (-1)^{p+1}C(\varphi_h(y_0))h^{p+1} + \mathcal{O}(h^{p+2}).$$

Since $\varphi_h(y_0) = y_0 + \mathcal{O}(h)$, we have

$$\Phi_h^*(y_0) - \varphi_h(y_0) = (y_0 - \Phi_{-h}(\varphi_h(y_0)))(I + \mathcal{O}(h)) = (-1)^p C(\varphi_h(y_0))h^{p+1} + \mathcal{O}(h^{p+2}).$$

[5] Symmetric method is sometimes regarded as a branch of geometric numerical integration methods [97, Chapter V].

Thus, the adjoint method has the following expansion

$$\Phi_h^*(y_0) = \varphi_h(y_0) + (-1)^p C(y_0) h^{p+1} + \mathcal{O}(h^{p+2}).$$

For a symmetric method $\Phi_h^* = \Phi_h$, this expansion implies $(-1)^p = 1$, which holds if and only if p is an even number. □

The following theorem characterises symmetric Runge–Kutta methods.

Theorem 2.4 (e.g., [99, Chapter II.8, Theorem 8.8])**.** If coefficients of an s-stage Runge–Kutta method satisfy

$$a_{s+1-i,s+1-j} + a_{ij} = b_j, \qquad i, j = 1, \dots, s, \tag{2.3}$$

then the Runge–Kutta method is symmetric.

Proof. It is checked that the adjoint method is also an s-stage Runge–Kutta method with coefficients

$$\begin{aligned} c_i^* &= 1 - c_{s+1-i}, \\ a_{ij}^* &= b_{s+1-j} - a_{s+1-i,s+1-j}, \\ b_j^* &= b_{s+1-j}. \end{aligned}$$

Hence, the original method is symmetric if

$$\begin{aligned} c_i &= 1 - c_{s+1-i}, \\ a_{ij} &= b_{s+1-j} - a_{s+1-i,s+1-j}, \\ b_j &= b_{s+1-j}. \end{aligned}$$

These conditions are all verified by the condition (2.3). □

There are several ways of realising high-order Runge–Kutta methods. In the subsequent subsections, we show two of them: collocation and composition approaches.

2.2.2 Collocation methods

Collocation methods are summarised. The idea is to approximate the exact solution by a polynomial so that the differential equation holds at several (i.e., finite) points.

Definition 2.7 (Collocation methods, e.g., [99, Chapter II.7])**.** Let $c_1, \dots, c_s$ be distinct real numbers $(0 \le c_1 < \cdots < c_s \le 1)$. The collocation polynomial $u(t)$ is a polynomial of degree s satisfying

$$\begin{aligned} &u(t_0) = y_0, \\ &\dot{u}(t_0 + c_i h) = f(t_0 + c_i h, u(t_0 + c_i h)), \qquad i = 1, \dots, s, \end{aligned}$$

and the numerical solution of the collocation method is defined by $y_1 = u(t_0 + h)$.

The connection between the Runge–Kutta and collocation methods is summarised in the following theorem.

Theorem 2.5 (e.g., Wright [197])**.** The collocation method is equivalent to the s-stage Runge–Kutta method with coefficients

$$a_{ij} = \int_0^{c_i} l_j(\tau)\,\mathrm{d}\tau, \qquad b_i = \int_0^1 l_i(\tau)\,\mathrm{d}\tau,$$

where $l_i(\tau)$ is the Lagrange polynomial $l_i(\tau) = \prod_{l \ne i} (\tau - c_l)/(c_i - c_l)$.

Proof. Let $k_i := \dot{u}(t_0 + c_i h)$. Then $\dot{u}$ can be expressed as

$$\dot{u}(t_0 + \tau h) = \sum_{j=1}^{s} k_j \cdot l_j(\tau).$$

Integrating this expression with respect to τ from 0 to c_i leads to

$$u(t_0 + c_i h) = y_0 + h \sum_{j=1}^{s} k_j \int_0^{c_i} l_j(\tau)\,\mathrm{d}\tau.$$

□

Let $c_1, \ldots, c_s$ be the zeros of the s-th shifted Legendre polynomial

$$\frac{\mathrm{d}^s}{\mathrm{d}x^s}(x^s(x-1)^s).$$

In this case, the collocation methods have order $p = 2s$ and are called Gauss methods. Here are the first three examples.

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} \qquad \begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \qquad \begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}$$

2.2.3 Composition methods

One can construct a higher-order integrator by composing low-order integrators with different step-sizes. The idea of the composition methods was mainly developed by Yoshida [215], Suzuki [179] and Mclachlan [140].

Definition 2.8 (Composition methods, e.g., [97, Chapter II.4])**.** Let Φ_h be a basic one-step method, and $\gamma_1, \ldots, \gamma_s$ real numbers. We call its composition with stepsizes $\gamma_1 h, \ldots, \gamma_s h$, i.e.,

$$\Psi_h = \Phi_{\gamma_s h} \circ \cdots \circ \Phi_{\gamma_1 h},$$

the corresponding composition method.

We regard Ψ_h as a new one-step method. The order of Ψ_h becomes larger than that of the basic method Φ_h if the parameters $\gamma_1, \ldots, \gamma_s$ are selected such that they satisfy the assumptions in the following theorem.

Theorem 2.6 (e.g., [97, Chapter II.4])**.** Let Φ_h be a one-step method of order p. If

$$\gamma_1 + \cdots + \gamma_s = 1, \tag{2.4}$$

$$\gamma_1^{p+1} + \cdots + \gamma_s^{p+1} = 0, \tag{2.5}$$

the corresponding composition method Ψ_h is at least of order $p + 1$, in the sense that $\|y(t_0 + h) - \Psi_h(y_0)\| = \mathcal{O}(h^{p+2})$.

Proof. Assume that Φ_h is of order p and has the following expansion

$$\Phi_h(y_0) = \varphi_h(y_0) + C(y_0)h^{p+1} + \mathcal{O}(h^{p+2}),$$

where φ_h denotes the exact flow and $C(\neq 0)$ is a smooth function. We write $z_0 = y_0$ and denote $\Phi_{\gamma_i h} \circ \cdots \circ \Phi_{\gamma_1 h}(y_0)$ by z_i (hence, $y_1 = z_s$). By the assumption, it follows that

$$\begin{aligned} e_i &:= \Phi_{\gamma_i h}(z_{i-1}) - \varphi_{\gamma_i h}(z_{i-1}) \\ &= C(z_{i-1})\gamma_i^{p+1}h^{p+1} + \mathcal{O}(h^{p+2}). \end{aligned}$$

Moreover,

$$\begin{aligned} E_i &:= \varphi_{(\gamma_{i+1}+\cdots+\gamma_s)h}\big(\Phi_{\gamma_i h}(z_{i-1})\big) - \varphi_{(\gamma_{i+1}+\cdots+\gamma_s)h}\big(\varphi_{\gamma_i h}(z_{i-1})\big) \\ &= (I + \mathcal{O}(h))e_i. \end{aligned}$$

Note that $E_s = e_s$, because $z_i = y_0 + \mathcal{O}(h)$, $C(z_i) = C(y_0) + \mathcal{O}(h)$. Consequently, we get

$$\Psi_h(y_0) - \varphi_h(y_0) = \sum_{i=1}^{s} E_i = C(y_0)(\gamma_1^{p+1} + \cdots + \gamma_s^{p+1})h^{p+1} + \mathcal{O}(h^{p+2}) = \mathcal{O}(h^{p+2}).$$

□

We here mention a simple but important example. Note that the equations (2.4) and (2.5) have no real solutions for odd p. Let us consider a symmetric one-step method of order p (even number) as a basic method Φ_h. By setting $s = 3$ and imposing $\gamma_1 = \gamma_3$, we obtain the solutions of (2.4) and (2.5):

$$\gamma_1 = \gamma_3 = \frac{1}{2 - 2^{1/(p+1)}}, \qquad \gamma_2 = \frac{2^{1/(p+1)}}{2 - 2^{1/(p+1)}}. \tag{2.6}$$

In this case, the corresponding composition method is also symmetric and thus of order $p+2$. This idea is called *the triple jump*.

The composition method can be applied to not only Runge–Kutta methods but also every one-step method. It is also worth mentioning that the composition of a Runge–Kutta method is also another Runge–Kutta method [28]. In general, the composition of a B-series integrator is a B-series integrator [100].

The composition methods have several advantages.

- In general, it is cumbersome to implement higher-order integrators. However, the implementation of the composition methods is relatively easy: one only have to call a low-order integrator several times. For example, by beginning with the midpoint rule or trapezoidal rule, one can obtain an integrators of arbitrary high-order.
- The composition methods usually share several properties of the basic method. For example, if the basic method preserves the energy of the problem, then the composition method also inherits the energy-preservation.

On the other hand, the composition methods have the following drawbacks.

- The computational cost would be a big deal. For example, we have to call a symmetric, second-order integrator 3^{s-1} times for an integrator of order $p = 2s$.
- Composition methods usually contain parameters whose absolute value is larger than 1 (e.g., $|\gamma_2| > 1$ for the triple jump (2.6)). Such a parameter might deteriorate the stability. This becomes pronounced as the order increases.
- Every composition method includes at least one negative parameter (see (2.5))[6]. Therefore, the composition methods are unfit for problems like dissipative systems that are not time-symmetric.

[6] Composition methods with complex-valued coefficients with positive real parts were recently considered in [12, 13].

2.2.4 Partitioned Runge–Kutta methods

We consider a partitioned system of differential equations

$$\dot{y} = f(y,z),$$
$$\dot{z} = g(y,z).$$

Instead of applying the same Runge–Kutta method to both equations, different Runge–Kutta methods can be applied. That is, we integrate the first system by (a,b,c) and the second by $(\hat{a},\hat{b},\hat{c})$:

$$Y_i = y_0 + h\sum_{j=1}^{s} a_{ij} f(Y_j,Z_j), \qquad i=1,\dots,s,$$
$$Z_i = z_0 + h\sum_{j=1}^{s} \hat{a}_{ij} g(Y_j,Z_j), \qquad i=1,\dots,s,$$
$$y_1 = y_0 + h\sum_{i=1}^{s} b_i f(Y_i,Z_i),$$
$$z_1 = z_0 + h\sum_{i=1}^{s} \hat{b}_i g(Y_i,Z_i).$$

A numerical method of this type is called a partitioned Runge–Kutta method. In this thesis, the idea of partitioning will be used only for Poisson systems in Sections 2.3.2, 2.5.5 and 3.5.

The concept of B-series can be extended to partitioned systems. Below the so called P-series is briefly summarised [97, Chapter III.2].

Let

$$TP = \{\bullet, \circ, \dots\}$$

be a set of rooted bi-coloured trees. This set and some mappings are defined by analogy with those for B-series. First, the above set of rooted bi-coloured trees is recursively defined as follows:

(a) the graph $\bullet$ and $\circ$ belong to TP;

(b) if $\tau_1,\dots,\tau_m \in TP$, then the graph obtained by connecting the roots of $\tau_1,\dots,\tau_m$ to a new common root $\bullet$ also belongs to TP. The new tree is denoted by $\tau = [\tau_1,\dots,\tau_m]_y$. Similarly, the new tree whose root is $\circ$ is denoted by $\tau = [\tau_1,\dots,\tau_m]_z$.

Next, the symmetry coefficient $\sigma : TP \to \mathbb{R}$ is defined recursively by

$$\sigma(\bullet) = \sigma(\circ) = 1, \qquad \sigma(\tau) = \sigma(\tau_1)\cdots\sigma(\tau_m)\mu_1!\cdots\mu_m!,$$

for $\tau = [\tau_1,\dots,\tau_m]_{y\,\mathrm{or}\,z}$, where the integer μ_i denotes the number of the equal trees of τ_i. Other mappings are illustrated in Table 2.3.

For a mapping $a : TP \cup \{\emptyset_y,\emptyset_z,\} \to \mathbb{R}$, a formal series of the form

$$P(a,(y,z)) = \begin{pmatrix} a(\emptyset_y)y + \displaystyle\sum_{\tau\in TP_y} \frac{h^{|\tau|}}{\sigma(\tau)} F(\tau)(y,z) \\ a(\emptyset_z)z + \displaystyle\sum_{\tau\in TP_z} \frac{h^{|\tau|}}{\sigma(\tau)} F(\tau)(y,z) \end{pmatrix}$$

is called a P-series.

Table 2.3: Bi-coloured trees and their related mappings.

$\|\tau\|$	τ	tree	$F(\tau)$	$\sigma(\tau)$	$e(\tau)$	$\phi(\tau)$
1	$\bullet$		f	1	1	$\sum_i b_i$
2	$[\bullet]_y$		$f_y f$	1	1/2	$\sum_{i,j} b_i a_{ij}$
2	$[\circ]_y$		$f_z g$	1	1/2	$\sum_{i,j} b_i \hat{a}_{ij}$
3	$[\bullet,\bullet]_y$		$f_{yy}(f,f)$	2	1/3	$\sum_{ijk} b_i a_{ij} a_{ik}$
3	$[\bullet,\circ]_y$		$f_{yz}(f,g)$	1	1/3	$\sum_{ijk} b_i a_{ij} \hat{a}_{ik}$
3	$[\circ,\circ]_y$		$f_{zz}(g,g)$	2	1/3	$\sum_{ijk} b_i \hat{a}_{ij} \hat{a}_{ik}$
3	$[[\bullet]_y]_y$		$f_y f_y f$	1	1/6	$\sum_{ijk} b_i a_{ij} a_{jk}$
3	$[[\circ]_y]_y$		$f_y f_z g$	1	1/6	$\sum_{ijk} b_i a_{ij} \hat{a}_{jk}$
3	$[[\bullet]_z]_y$		$f_z g_y f$	1	1/6	$\sum_{ijk} b_i \hat{a}_{ij} a_{jk}$
3	$[[\circ]_z]_y$		$f_z g_z g$	1	1/6	$\sum_{ijk} b_i \hat{a}_{ij} \hat{a}_{jk}$
1	$\circ$		g	1	1	$\sum_i \hat{b}_i$
2	$[\bullet]_z$		$g_y f$	1	1/2	$\sum_{i,j} \hat{b}_i a_{ij}$
2	$[\circ]_z$		$g_z g$	1	1/2	$\sum_{i,j} \hat{b}_i \hat{a}_{ij}$

Theorem 2.7 (e.g., [99])**.** The partitioned Runge–Kutta method is of order p, i.e., $y_1 - y(t_0+h) = \mathcal{O}(h^{p+1})$ and $z_1 - z(t_0+h) = \mathcal{O}(h^{p+1})$, if and only if

$$\phi(\tau) = e(\tau) \qquad \text{for} \qquad |\tau| \le p.$$

Theorem 2.8 (e.g., [97, Chapter V.2.2])**.** If both Runge–Kutta methods (a, b, c) and $(\hat{a}, \hat{b}, \hat{c})$ are symmetric (see Theorem 2.4), then the corresponding partitioned Runge–Kutta method is also symmetric.

2.3 Hamiltonian mechanics

Here, we give a short introduction of Hamiltonian mechanics based on [3, 107, 132].

2.3.1 Hamiltonian systems

We briefly overview Hamiltonian mechanics. Firstly, we derive Hamiltonian systems starting with Hamilton's principle of stationary action, and show some examples. We then summarise some geometric properties of Hamiltonian systems.

We denote generalised coordinates of position of a mechanical system with d degrees of freedom by $q = (q_1, \dots, q_d)^\top$, and introduce Lagrangian $L(q, \dot{q})$. The variational principle of Hamilton states

$$\delta\mathcal{L} = 0, \qquad \mathcal{L} = \int_{t_0}^{t_1} L(q, \dot{q})\,\mathrm{d}t,$$

with fixed start- and end-points $q_1 = q(t_1)$ and $q_2 = q(t_2)$. This principle means that $q(t)$ between $q_1 = q(t_1)$ and $q_2 = q(t_2)$ evolves in such a way that it is a stationary point of the Lagrangian $\mathcal{L}$. Taking the functional derivative for variations $\delta q : [t_0, t_1] \to \mathbb{R}^d$ with vanishing boundary condition

$\delta q(t_0) = \delta q(t_1) = 0$, we have

$$\begin{aligned}
0 &= \frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathcal{L}[q+\epsilon\delta q]\bigg|_{\epsilon=0} \\
&= \frac{\mathrm{d}}{\mathrm{d}\epsilon}\int_{t_0}^{t_1} L(q+\epsilon\delta q, \dot{q}+\epsilon\delta\dot{q})\,\mathrm{d}t\bigg|_{\epsilon=0} \\
&= \int_{t_0}^{t_1}\left(\frac{\partial L}{\partial q}(q+\epsilon\delta q, \dot{q}+\epsilon\delta\dot{q})\delta q + \frac{\partial L}{\partial \dot{q}}(q+\epsilon\delta q, \dot{q}+\epsilon\delta\dot{q})\delta\dot{q}\right)\mathrm{d}t\bigg|_{\epsilon=0} \\
&= \int_{t_0}^{t_1}\left(\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}}\right)\delta q\,\mathrm{d}t + \left[\frac{\partial L}{\partial \dot{q}}\delta q\right]_{t_0}^{t_1} \\
&= \int_{t_0}^{t_1}\left(\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}}\right)\delta q\,\mathrm{d}t.
\end{aligned}$$

Since the above relation holds for all variations δq, Hamilton's principle is equivalent to the Euler–Lagrange equation

$$\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}} = 0. \tag{2.7}$$

For classical mechanics, the Lagrangian L often has the form of kinetic minus potential energy:

$$L(q,\dot{q}) = T(q,\dot{q}) - U(q), \qquad T(q,\dot{q}) = \frac{1}{2}\dot{q}^\top M(q)\dot{q},$$

where $M(q)$ is a symmetric, positive definite matrix.

In order to turn to the Hamiltonian formulation, we introduce the conjugate momenta

$$p_i = \frac{\partial L}{\partial \dot{q}_i}, \qquad i = 1,\dots,d,$$

and the Hamiltonian

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

By using the chain rule, it is easy to show that

$$\frac{\partial H}{\partial p_i} = \dot{q}_i + \sum_{j=1}^{d}\left(p_j\frac{\partial \dot{q}_j}{\partial p_i} - \frac{\partial L}{\partial \dot{q}_j}\frac{\partial \dot{q}_j}{\partial p_i}\right) = \dot{q}_i$$

and

$$\frac{\partial H}{\partial q_i} = \sum_{j=1}^{d} p_j\frac{\partial \dot{q}_j}{\partial q_i} - \frac{\partial L}{\partial q_i} - \sum_{j=1}^{d}\frac{\partial L}{\partial \dot{q}_j}\frac{\partial \dot{q}_j}{\partial q_i} = -\frac{\partial L}{\partial \dot{q}} = -\dot{p}_i.$$

Thus, the Euler–Lagrange equation is equivalent to Hamilton's equations

$$\begin{aligned}
\dot{q}_i &= \frac{\partial H}{\partial p_i}, \qquad i = 1,\dots,d, \\
\dot{p}_i &= -\frac{\partial H}{\partial q_i}, \qquad i = 1,\dots,d.
\end{aligned}$$

By introducing the coordinates $y = (q_1, \dots, q_d, p_1, \dots, p_d) \in \mathbb{R}^{2d}$, we often denote Hamilton's equations by the form

$$\dot{y} = J^{-1}\nabla H, \qquad J = \begin{pmatrix} O & -I \\ I & O \end{pmatrix} \tag{2.8}$$

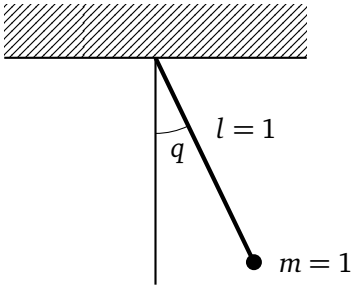
where J is a skew-symmetric constant matrix defined with the identity matrix $I \in \mathbb{R}^{d\times d}$ and zero matrix $O \in \mathbb{R}^{d\times d}$. We often refer to such a system of differential equations as a Hamiltonian system.

Example 2.1 (Simple pendulum)**.** A simple pendulum is described as the second-order equation

$$\ddot{q} + \sin q = 0,$$

when the mass $m = 1$, length $l = 1$ and the gravitation acceleration $g = 1$. This equation is equivalent to the Hamiltonian system with

$$H(q,p) = \frac{1}{2}p^2 - \cos q.$$



Example 2.2 (The Fermi–Pasta–Ulam problem)**.** The Fermi–Pasta–Ulam problem [75] is a simple model appearing in statistical mechanics. Due to its unexpected dynamical behaviour, this problem is regarded as a highly oscillatory test problem for numerical simulations. The following formulation is a modified version of the Fermi–Pasta–Ulam problem by Galgani et al. [84] (see also [97, Chapter I.5]).

As shown in Figure 2.2, we consider the motion of $2m$ mass points, connected with stiff linear and weak nonlinear springs alternately. Here, oscillations are caused by the stiff linear springs. When we consider springs with cubic nonlinearity, the motion is described by the Hamiltonian system with the Hamiltonian

$$H(q,p) = \frac{1}{2}\sum_{i=1}^{m}\left(p_{2i-1}^2 + p_{2i}^2\right) + \frac{\omega^2}{4}\sum_{i=1}^{m}(q_{2i} - q_{2i-1})^2 + \sum_{i=0}^{m}(q_{2i+1} - q_{2i})^4$$

where $p_i = \dot{q}_i$ and $\omega \gg 1$.

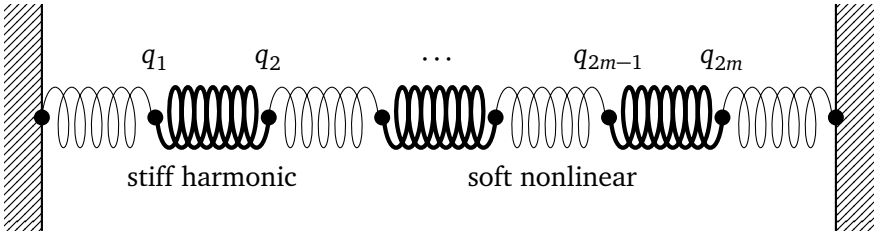


Figure 2.2: A variant of the Fermi–Pasta–Ulam problem (This figure is almost the same as Figure 5.1 in [97, Chapter I]). End points are fixed ($q_0 = q_{2m+1} = 0$).

Following [97, Chapter I.5], we introduce the coordinate transformation

$$x_{0,i} = \frac{q_{2i} + q_{2i-1}}{\sqrt{2}}, \qquad x_{1,i} = \frac{q_{2i} - q_{2i-1}}{\sqrt{2}},$$
$$y_{0,i} = \frac{p_{2i} + p_{2i-1}}{\sqrt{2}}, \qquad y_{1,i} = \frac{p_{2i} - p_{2i-1}}{\sqrt{2}},$$

so that the Hamiltonian becomes

$$H(y,x) = \frac{1}{2}\sum_{i=1}^{m}\left(y_{0,i}^2 + y_{1,i}^2\right) + \frac{\omega^2}{2}\sum_{i=1}^{m} x_{1,i}^2 + \frac{1}{4}\left(\left(x_{0,1} - x_{1,1}\right)^4 + \sum_{i=1}^{m-1}\left(x_{0,i+1} - x_{1,i+1} - x_{0,i} - x_{1,i}\right)^4 + \left(x_{0,m} + x_{1,m}\right)^4\right).$$

This system also nearly preserves the oscillatory energy defined by

$$I = \sum_{i=1}^{m} I_i, \qquad \text{where} \qquad I_i = \frac{1}{2}\left(y_{1,i}^2 + \omega^2 x_{1,i}^2\right).$$

More precisely, $I(t) = I(0) + \mathcal{O}(\omega^{-1})$ holds [97, Chapter XIII]. In Figure 2.3, Hamiltonian and oscillatory energies of the exact solution are plotted. There is an exchange of oscillatory energies of the scale $\mathcal{O}(1)$. Moreover, oscillations of the scale $\mathcal{O}(\omega^{-1})$ are observed. This figure indicates that while the oscillatory energy of each stiff spring is slowly transferred to other springs, each spring itself oscillates faster.

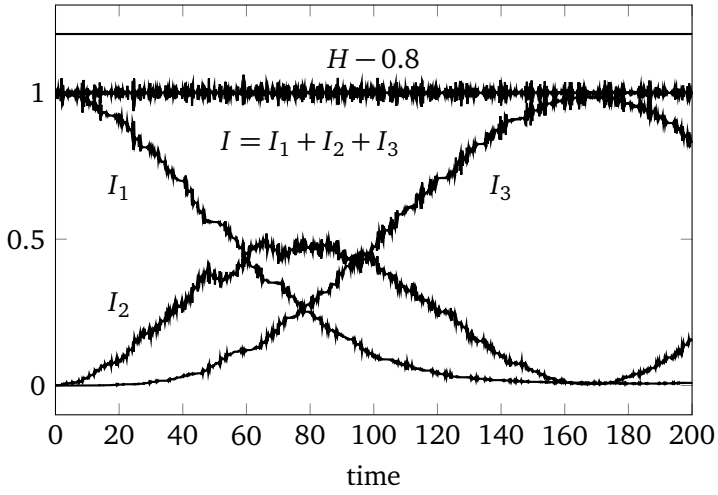


Figure 2.3: The Hamiltonian and oscillatory energies ($H - 0.8$, I, I_1, I_2, I_3) for the exact solution of the Fermi–Past–Ulam problem are plotted. The parameters are set to $m = 3$ and $\omega = 50$. The initial values are set to $x_{0,1}(0) = 1$, $y_{0,1}(0) = 1$, $x_{1,1}(0) = \omega^{-1}$, $y_{1,1}(0) = 1$ and zero for other components. For the Hamiltonian, $H - 0.8$ is plotted just to save space.

We now give some basic properties of Hamiltonian systems. We denote the exact flow of the system (2.8) by $\varphi_H^t : \mathbb{R}^{2d} \ni y(t_0) \mapsto y(t_0 + t) \in \mathbb{R}^{2d}$.

Theorem 2.9 (Symplecticity, e.g., [97, Chapter VI.2])**.** For all t, the flow map $y \mapsto \varphi_H^t(y)$ is a symplectic transformation, i.e., the map satisfies

$$\left(\frac{\partial \varphi_H^t}{\partial y}\right)^{\top} J^{-1} \left(\frac{\partial \varphi_H^t}{\partial y}\right) = J^{-1}. \tag{2.9}$$

Proof. Let $F(t) = \partial \varphi_H^t / \partial y$. By differentiating $F(t)$ with respect to t, we obtain the variational equation $\dot{F} = J^{-1}\nabla^2 H(\varphi_H^t(y))F$, where $\nabla^2 H(q,p)$ is a Hessian matrix of $H(q,p)$, with the initial condition $F(0) = I_{2d}$ (identity matrix of size $2d$). We aim to prove $F(t)^{\top} J^{-1} F(t) = J^{-1}$. Since this obviously holds when

$t = 0$, we only have to show $\frac{\mathrm{d}}{\mathrm{d}t}F(t)^\top J^{-1}F(t) = 0$.

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}F(t)^\top J^{-1}F(t) = \dot{F}^\top J^{-1}F + F^\top J^{-1}\dot{F}\\ &= \left(J^{-1}(\nabla^2 H)F\right)^\top J^{-1}F + F^\top J^{-1}J^{-1}(\nabla^2 H)F = F^\top(\nabla^2 H)^\top F - F^\top(\nabla^2 H)F = 0.\end{aligned}$$

The last equality follows from the symmetry of the Hessian matrix $\nabla^2 H$. □

The symplecticity (2.9) means that the sum of the oriented areas of the projections of a certain area in (q,p) onto (q_i,p_i) is preserved along the flow map. The simplest case, where $d = 1$ and the map is linear, is illustrated in Figure 2.4. We remark that although the exact flow of Hamiltonian systems also preserves a volume, the symplecticity is not equivalent to the volume preservation except for the case $d = 1$.

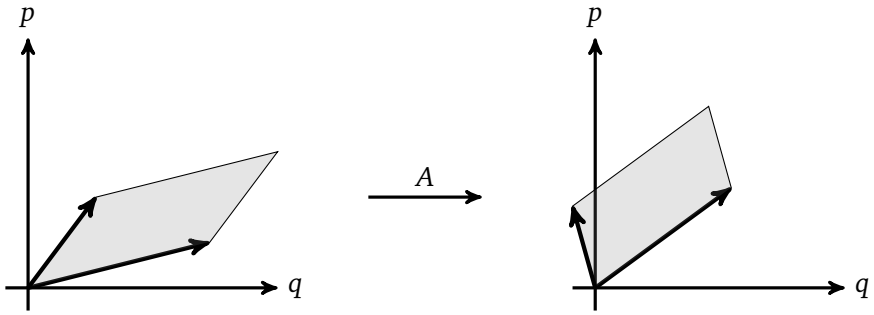


Figure 2.4: Illustration of symplecticity (area preservation when $d = 1$) of a linear mapping A.

The symplecticity can also be formulated in terms of differential forms.

Theorem 2.10 (e.g., [123, Chapter 3])**.** For all t, the flow map $y \mapsto \varphi_H^t(y)$ preserves the symplectic 2-form

$$\omega = \frac{1}{2}J\mathrm{d}y \wedge \mathrm{d}y = \mathrm{d}q \wedge \mathrm{d}p.$$

This is equivalent to Theorem 2.9.

Proof. Let $q(t+h) = \phi^q(q,p)$ and $p(t+h) = \phi^p(q,t)$, i.e., $\phi^q(q,p)$ and $\phi^p(q,p)$ denote the q and p part of $\varphi_H^t(y)$, respectively. Then, (2.9) is equivalent to

$$(\phi_q^q)^\top \phi_q^p = (\phi_q^p)^\top \phi_q^q, \tag{2.10}$$
$$(\phi_p^q)^\top \phi_p^p = (\phi_p^p)^\top \phi_p^q, \tag{2.11}$$
$$(\phi_p^p)^\top \phi_q^q - (\phi_p^q)^\top \phi_q^p = I,$$

because the left hand side of (2.9) is calculated to be

$$\begin{aligned}\left(\frac{\partial \varphi_H^t}{\partial y}\right)^\top J^{-1}\left(\frac{\partial \varphi_H^t}{\partial y}\right) &= \begin{pmatrix}(\phi_q^q)^\top & (\phi_q^p)^\top\\ (\phi_p^q)^\top & (\phi_p^p)^\top\end{pmatrix}\begin{pmatrix}O & I\\ -I & O\end{pmatrix}\begin{pmatrix}\phi_q^q & \phi_p^q\\ \phi_q^p & \phi_p^p\end{pmatrix}\\ &= \begin{pmatrix}(\phi_q^q)^\top\phi_q^p - (\phi_q^p)^\top\phi_q^q & (\phi_q^q)^\top\phi_p^p - (\phi_q^p)^\top\phi_p^q\\ (\phi_p^q)^\top\phi_q^p - (\phi_p^p)^\top\phi_q^q & (\phi_p^q)^\top\phi_p^p - (\phi_p^p)^\top\phi_p^q\end{pmatrix}.\end{aligned}$$

Note that (2.10) and (2.11) mean that $(\phi_q^p)^\top\phi_q^q$ and $(\phi_p^p)^\top\phi_p^q$ are symmetric matrices.

Since $\mathrm{d}\hat{q} = \phi_q^q \mathrm{d}q + \phi_p^q \mathrm{d}p$ and $\mathrm{d}\hat{p} = \phi_q^p \mathrm{d}q + \phi_p^p \mathrm{d}p$ with the notation $\hat{q} = q(t+h)$ and $\hat{p} = p(t+h)$, it follows

$$\begin{aligned}\mathrm{d}\hat{q} \wedge \mathrm{d}\hat{p} &= (\phi_q^q \mathrm{d}q + \phi_p^q \mathrm{d}p) \wedge (\phi_q^p \mathrm{d}q + \phi_p^p \mathrm{d}p) \\ &= \Big((\phi_q^p)^\top \phi_q^q \mathrm{d}q\Big) \wedge \mathrm{d}q + \Big(\big((\phi_p^p)^\top \phi_q^q - (\phi_p^q)^\top \phi_q^p\big)\mathrm{d}q\Big) \wedge \mathrm{d}p + \Big((\phi_p^p)^\top \phi_p^q \mathrm{d}p\Big) \wedge \mathrm{d}p.\end{aligned}$$

Therefore, $\mathrm{d}\hat{q} \wedge \mathrm{d}\hat{p} = \mathrm{d}q \wedge \mathrm{d}p$ if and only if (2.9) holds. □

Another well-known and important property of the flow is that it preserves the energy, i.e., Hamiltonian.

Theorem 2.11 (Energy-preservation)**.** For all t, the flow map $y \mapsto \varphi_H^t(y)$ preserves the energy in the sense that

$$H\big(\varphi_H^t(y)\big) = H(y).$$

Proof. The proof is straightforward.

$$\frac{\mathrm{d}}{\mathrm{d}t} H(y) = \nabla H(y)^\top \dot{y} = \nabla H(y)^\top J^{-1} \nabla H(y) = 0.$$

Note that the last equality is due to the skew-symmetry of J. □

2.3.2 Poisson systems

As a generalisation of Hamiltonian systems, Poisson systems play a crucial part in mathematical formulations for more complicated phenomena such as constrained mechanical systems and infinite dimensional mechanical systems. For more details, see, e.g., [97, Chapter VII.2].

Firstly we reformulate Hamiltonian systems using the Poisson bracket. A bracket of two functions $F(q,p)$ and $G(q,p)$ defined by

$$\{F,G\} = \sum_{i=1}^{d} \left(\frac{\partial F}{\partial q_i} \frac{\partial G}{\partial p_i} - \frac{\partial F}{\partial p_i} \frac{\partial G}{\partial q_i} \right)$$

is called the canonical Poisson bracket. It is easy to check that the Poisson bracket is bilinear and skew-symmetric

$$\{F,G\} = -\{G,F\}$$

and satisfies the Jacobi identity

$$\{\{F,G\},H\} + \{\{G,H\},F\} + \{\{H,F\},G\} = 0.$$

Every function $f = f(q,p)$ along the flow of a Hamiltonian system satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t} f(y(t)) = \{f,H\}(y(t))$$

because of the chain rule:

$$\frac{\mathrm{d}}{\mathrm{d}t} f(q(t),p(t)) = \sum_{i=1}^{d} \left(\frac{\partial f}{\partial q_i} \dot{q}_i + \frac{\partial f}{\partial p_i} \dot{p}_i \right) = \sum_{i=1}^{d} \left(\frac{\partial f}{\partial q_i} \frac{\partial H}{\partial p_i} - \frac{\partial f}{\partial p_i} \frac{\partial H}{\partial q_i} \right).$$

As the special case $f = y_i$, a Hamiltonian system

$$\dot{y}_i = \{y_i, H\}, \qquad i = 1, \dots, 2d$$

is recovered.

We now generalise the above idea to noncanonical cases. A skew-symmetric matrix $\Lambda(y) = \big(\lambda_{ij}(y)\big)$ is called a Poisson structure matrix if the Poisson bracket defined by

$$\{F, G\}(y) = \sum_{i,j=1}^{N} \frac{\partial F}{\partial y_i} \lambda_{ij}(y) \frac{\partial G}{\partial y_i}$$

satisfies the skew-symmetry

$$\{F, G\} = -\{G, F\}$$

and the Jacobi identity

$$\{\{F, G\}, H\} + \{\{G, H\}, F\} + \{\{H, F\}, G\} = 0.$$

As with the canonical case, given a Hamiltonian $H(y)$, the motion is governed by

$$\frac{\mathrm{d}}{\mathrm{d}t} f(y(t)) = \{f, H\}(y(t)).$$

By taking $f = y_i$, we obtain the system of equations

$$\dot{y}(t) = \Lambda(y) \nabla H(y),$$

which is called a *Poisson system*.

Below, we list two examples.

Example 2.3 (Lotka–Volterra equations)**.** The Lotka–Volterra equations describe dynamics in biological systems, in which two species interact each other. The systems is formulated as

$$\dot{u} = u(v - \alpha), \qquad \dot{p} = v(\beta - u),$$

where u and v denote the number of prey and predators, respectively. This system can be rewritten as the Poisson system

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & uv \\ -uv & 0 \end{pmatrix} \nabla H(u, v), \qquad H(u, v) = u - \beta \ln u + v - \alpha \ln v.$$

Example 2.4 (Euler equations)**.** The motion of a rigid body under no forces is described by the Euler equation

$$\dot{q} = ((\alpha - \beta) q_2 q_3, (1 - \alpha) q_3 q_1, (\beta - 1) q_1 q_2)^{\top}.$$

This system can be seen as the Poisson system

$$\dot{q} = \begin{pmatrix} 0 & \alpha q_3 & -\beta q_2 \\ -\alpha q_3 & 0 & q_1 \\ \beta q_2 & -q_1 & 0 \end{pmatrix} \nabla H(q), \qquad H(q) = \frac{q_1^2 + q_2^2 + q_3^2}{2}.$$

2.4 Symplectic methods

This section summarises symplectic integration methods. A numerical one-step method is said to be symplectic if the solution satisfies

$$\mathrm{d}q_n \wedge \mathrm{d}p_n = \mathrm{d}q_0 \wedge \mathrm{d}p_0.$$

2.4.1 First examples

Firstly, we give a quick review on symplectic methods through the simplest example.

Theorem 2.12 (de Vogelaere [195])**.** The so called symplectic Euler methods

$$\begin{aligned} q_{n+1} &= q_n + hH_p(q_n, p_{n+1}), \\ p_{n+1} &= p_n - hH_q(q_n, p_{n+1}), \end{aligned}$$

or

$$\begin{aligned} q_{n+1} &= q_n + hH_p(q_{n+1}, p_n), \\ p_{n+1} &= p_n - hH_q(q_{n+1}, p_n) \end{aligned}$$

are symplectic and of order 1.

Proof. The first order convergence is obvious. We shall prove the symplecticity for the first method. It immediately follows that

$$\begin{aligned} \mathrm{d}q_{n+1} &= \mathrm{d}q_n + h\big(H_{pq}\mathrm{d}q_n + H_{pp}\mathrm{d}p_{n+1}\big), \\ \mathrm{d}p_{n+1} &= \mathrm{d}p_n - h\big(H_{qq}\mathrm{d}q_n + H_{qp}\mathrm{d}p_{n+1}\big). \end{aligned}$$

Note that the matrices H_{qq} and H_{pp} are symmetric, and $H_{qp} = H_{pq}^{\top}$. Taking the wedge product with $\mathrm{d}p_{n+1}$ from the right to the first equation and $\mathrm{d}q_n$ from the left to the second equation, we obtain

$$\begin{aligned} \mathrm{d}q_{n+1} \wedge \mathrm{d}p_{n+1} &= \mathrm{d}q_n \wedge \mathrm{d}p_{n+1} + hH_{pq}\mathrm{d}q_n \wedge \mathrm{d}p_{n+1}, \\ \mathrm{d}q_n \wedge \mathrm{d}p_{n+1} &= \mathrm{d}q_n \wedge \mathrm{d}p_n - hH_{qp}\mathrm{d}q_n \wedge \mathrm{d}p_{n+1}. \end{aligned}$$

Then, we readily have $\mathrm{d}q_{n+1} \wedge \mathrm{d}p_{n+1} = \mathrm{d}q_n \wedge \mathrm{d}p_n$. □

The symplectic Euler methods are implicit in general. For separable Hamiltonian systems $H(q,p) = T(p) + U(q)$, however, they become explicit.

When $T(p) = p^2/2$, a Hamiltonian system can be written componentwise

$$\dot{q} = p, \qquad \dot{p} = -\nabla U(q),$$

or a second-order differential equation

$$\ddot{q} = -\nabla U(q).$$

The most natural discretisation of the latter equation is

$$q_{n+1} - 2q_n + q_{n-1} = -h^2 \nabla U(q), \tag{2.12}$$

which is called the Störmer–Verlet method. This method can be reformulated as a one-step method

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2}\nabla U(q_n), \\ q_{n+1} &= q_n + h p_{n+1/2}, \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2}\nabla U(q_{n+1}). \end{aligned}$$

The Sörmer–Verlet method is symepectic, and moreover, in contrast to the symplectic Euler methods, symmetric and thus of order two.

Some symplectic methods themselves had been found in the early twentieth century, long before the concept of symplecticity became of interest. The Störmer–Verlet method is said to be the oldest one, which was first constructed by Störmer [176] in the context of astronomy in 1907, and rediscovered independently by Verlet [194] in the context of molecular dynamics in 1967.

2.4.2 Symplectic Runge–Kutta methods

Higher-order symplectic methods can be constructed by the composition method with second-order symplectic methods. In addition, there are other ways to realise such methods. This subsection reviews symplectic Runge–Kutta methods.

In 1988, a condition of Runge–Kutta methods being symplectic was obtained independently by Lasagni [122], Sanz-Serna [173] and Suris [178].

Theorem 2.13 ([122, 173, 178])**.** A Runge–Kutta method solving Hamiltonian systems is symplectic if the following conditions are satisfied

$$b_i a_{ij} + b_j a_{ji} = b_i b_j, \qquad 1 \le i, j \le s. \tag{2.13}$$

Although this theorem can be proved in a more direct way, we here follows the approach by Boschev–Scovel [14]. Their proof is based on the following theorem.

Theorem 2.14 ([14])**.** If a Runge–Kutta method conserves quadratic first integrals (i.e., for any differential equations $\dot{y} = f(y)$ with a quadratic first integral $I(y) = y^\top Q y$ (Q is a symmetric matrix), $I(y_1) = I(y_0)$ holds), then it is symplectic.

Proof. For Runge–Kutta methods, the following diagram commutes:

$$\begin{array}{ccc} \dot{y} = f(y),\ y(0) = y_0 & \xrightarrow{\text{RK method}} & \{y_n\} \\ \downarrow \frac{\partial}{\partial y_0} & & \downarrow \frac{\partial}{\partial y_0} \\ \dot{y} = f(y),\ y(0) = y_0 & \xrightarrow{\text{RK method}} & \{y_n, \Psi_n\} \\ \dot{\Psi} = f'(y)\Psi(y),\ \Psi(0) = I & & \end{array}$$

Note that for the Hamiltonian system (i.e., $\dot{y} = J^{-1}\nabla H(y)$), its variational equation is

$$\dot{\Psi} = J^{-1}\nabla^2 H(y)\Psi.$$

Since

$$(J^{-1}\nabla^2 H(y)\Psi)^\top J\Psi + \Psi^\top J(J^{-1}\nabla^2 H(y)\Psi) = 0,$$

the diagram indicates that $\Psi^\top J\Psi$ is a quadratic first integral of the variational equation.

Therefore every Runge–Kutta method that preserves quadratic first integrals is a symplectic method. □

Conversely, symplectic Runge–Kutta methods are usually constructed so that they preserve quadratic first integrals.

Proof of Theorem 2.13. Let us assume that $\dot{y} = f(y)$ has a quadratic invariant $\langle y, Qy\rangle (:= y^\top Qy) =$ const., which means $\langle v, Qf(v)\rangle = 0$ for all v. For the internal stages of the Runge–Kutta method

$$Y_i = y_0 + h\sum_{j=1}^{s} a_{ij} f(Y_j),$$

it follows from $\langle Y_i, Qf(Y_i)\rangle = 0$ that

$$0 = \langle Y_i, Qf(Y_i)\rangle = \langle y_0, Qf(Y_i)\rangle + h\sum_{j=1}^{s} a_{ij}\langle f(Y_j), Qf(Y_i)\rangle.$$

Similarly, for the final stage

$$y_1 = y_0 + h\sum_{i=1}^{s} b_i f(Y_i),$$

it follows that

$$\langle y_1, Qy_1\rangle = \langle y_0, Qy_0\rangle + h\sum_{i=1}^{s} b_i\langle y_0, Qf(Y_i)\rangle + h\sum_{j=1}^{s} b_j\langle y_0, Qf(Y_j)\rangle + h^2\sum_{i,j=1}^{s} b_i b_j\langle f(Y_i), Qf(Y_j)\rangle.$$

Therefore $\langle y_1, Qy_1\rangle = \langle y_0, Qy_0\rangle$ if the conditions (2.13) are satisfied. □

Remark 2.1. A similar condition for the symplecticity of Runge–Kutta methods had been already known as a condition of *B-stability*[7] of Runge–Kutta methods [25, 30, 60]. Let us consider a nonlinear equation $\dot{y}(t) = f(t, y(t))$ with the property

$$\langle u - v, f(t,u) - f(t,v)\rangle \le 0.$$

Two solutions y and $\tilde{y}$ starting from different initial values satisfy

$$\frac{\mathrm{d}}{\mathrm{d}t}\|y(t) - \tilde{y}(t)\| \le 0.$$

A Runge–Kutta method is said to be B-stable if

$$\|y_{n+1} - \tilde{y}_{n+1}\| \le \|y_n - \tilde{y}_n\|.$$

If the matrix M with the elements

$$m_{ij} = b_i a_{ij} + b_j a_{ji} - b_i b_j$$

is positive semi-definite, and $b_i \ge 0$ $(i = 1, \ldots, s)$, the corresponding Runge–Kutta method is B-stable.

[7] The concept of B-stability was first introduced by Butcher [30] and in that paper it was already referred to as the B-stability. According to Butcher's book [32, p. 250], it was named "because it is one step more stringent than A-stability."

2.4.3 Backward error analysis

When we consider the error estimate of a numerical method, we usually estimate the local error $\|y(t_0+h)-y_1\|$ or the global error $\|y(t_0+nh)-y_n\|$. Such analyses are based on the Taylor expansion and called forward error analyses. On the other hand, We can consider at least formally a differential equation whose solution coincides with the numerical solution. The idea of *backward error analysis* is to search for a differential equation of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{y} = f(\tilde{y}) + hf_1(\tilde{y}) + h^2 f_2(\tilde{y}) + h^3 f_3(\tilde{y}) + \cdots, \qquad \tilde{y}(t_0) = y_0$$

such that $y_n = \tilde{y}(t_0 + nh)$, and to analyse the above modified equation. If the method is of order p, we have $f_j = 0$ $(j = 1, \ldots, p-1)$.

In Chapter 1, we saw that the difference between numerical and exact energies are bounded for a symplectic method over a long-time interval. This property is verified by the backward error analysis.

Theorem 2.15 (e.g., [97, Chapter IX, Theorem 3.1])**.** The modified equation of a symplectic integrator applied to a sufficiently smooth Hamiltonian system is also a Hamiltonian system. More precisely, the modified equation can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{y} = J^{-1}\nabla\tilde{H}(\tilde{y})$$

with the modified Hamiltonian

$$\tilde{H}(\tilde{y}) = H(\tilde{y}) + hH_1(\tilde{y}) + h^2 H_2(\tilde{y}) + h^3 H_3(\tilde{y}) + \cdots. \tag{2.14}$$

This theorem indicate that a symplectic method of order p preserves

$$\tilde{H}(\tilde{y}) = H(\tilde{y}) + h^p H_p(\tilde{y}) + h^{p+1} H_{p+1}(\tilde{y}) + \cdots,$$

and thus $|H(y_n) - H(y_0)| \le \mathcal{O}(h^p)$.

Remark 2.2. Note that the expression of the modified Hamiltonian (2.14) does not converge in general. Strictly speaking, we terminate the expansion after some finite number of terms, and have to check if the corresponding approximation is valid for an exponentially long-time interval. See [97, Chapter IX] for a more detailed discussion.

Sketch of proof of Theorem 2.15. The proof is by induction. We assume $f_j = J^{-1}\nabla H_j$ for $j = 1, \ldots, r$ and prove the existence of H_{r+1} so that $f_{r+1} = J^{-1}\nabla H_{r+1}$.

Consider the truncated modified equation

$$\dot{\tilde{y}} = f(\tilde{y}) + hf_1(\tilde{y}) + \cdots + h^r f_r(\tilde{y}),$$

and denote the exact flow of this truncated equation by $\varphi_{r,t}(y_0)$. It follows from the Taylor series expansion that

$$\begin{aligned}
\Phi_h(y_0) &= \varphi_{r,h}(y_0) + h^{r+2} f_{r+1}(y_0) + \mathcal{O}(h^{r+3}), \\
\Phi_h'(y_0) &= \varphi_{r,h}'(y_0) + h^{r+2} f_{r+1}'(y_0) + \mathcal{O}(h^{r+3}).
\end{aligned}$$

Since Φ_h and $\varphi_{r,h}$ are both symplectic maps and $\varphi_{r,h}'(y_0) = I + \mathcal{O}(h)$, we have

$$J^{-1} = \Phi_h'(y_0)^\top J^{-1} \Phi_h'(y_0) = J^{-1} + h^{r+2}\big(f_{r+1}'(y_0)^\top J^{-1} + J^{-1} f_{r+1}'(y_0)\big) + \mathcal{O}(h^{r+3}),$$

which indicates that $J^{-1} f_{r+1}'(y_0)$ is symmetric. Hence, there exists H_{r+1} such that $f_{r+1} = J^{-1}\nabla H_{r+1}$ because of the integrability lemma (see [97, Chapter VI, Theorem 2.7]): briefly speaking, if f' is symmetric, there exists H such that $f = \nabla H$. □

2.5 Energy-preserving methods

This section summarises energy-preserving integration methods. A numerical one-step method is said to be energy-preserving if the solution satisfies

$$H(y_n) = H(y_0).$$

Energy-preserving methods are relatively new compared with symplectic methods. There are several reasons, one of which is shown in the following theorem.

Theorem 2.16 (e.g., Celledoni et al. [47])**.** No Runge–Kutta method is energy-preserving in general.

Proof. Let us consider the case $H(q,p) = p - F(q)$, i.e.,

$$\dot{q} = 1, \qquad \dot{p} = f(q), \qquad \big(f(q) = F'(q)\big).$$

Every Runge–Kutta method with the property $\sum_{i=1}^{s} b_i = 1$ provides an exact solution for the variable q, i.e., $q_1 = q_0 + h$. Thus, the energy-preservation requires that p_1 also coincides with the exact solution $p(t_0+h) = p_0 + \int_{t_0}^{t_0+h} f(t)\,\mathrm{d}t$. However, this is impossible in general (this can be verified by considering an f that is 0 at the quadrature nodes but has non-zero integral). □

Although Runge–Kutta methods cannot be energy-preserving, energy-preserving integrators can be constructed relatively easily by simple approaches: projection methods (see [4, 6, 9, 62, 70, 92, 93] and references therein) and methods on local coordinates [164, 165]. See also [97, Section IV.4 and 5]. However, more sophisticated approaches such as the discrete gradient method have been developed in the last two decades. They are reviewed in the rest of this section.

Remark 2.3. In Part I, we promise that by energy-preserving integrators we mean we target only one first integral. However, some Hamiltonian systems have more than one first integrals, and indeed, numerical methods preserving more than one first integrals have been considered. For example, the Kepler problem also preserves the so called angular momentum and Runge–Lenz vector. Energy-preserving integrators which also inherit such invariants are proposed by Brugnano–Iavernaro [21], Brugnano–Sun [24], Dahlby et al. [62], Kozlov [119] and Minesaki–Nakamura [143], for example.

Remark 2.4. Although we skip detailed explanations of the projection methods and methods on local coordinates, they are important concepts in the context of numerical methods for differential equations on manifolds. For example, Lie group integrators for differential equations on Lie groups have been developed in the last two decades (reviews are found in [46, 114]).

2.5.1 Discrete gradient method for Hamiltonian systems

The discrete gradient method is summarised.

Definition 2.9 (Discrete gradient)**.** Let $N \in \mathbb{Z}^+$ and $H : \mathbb{R}^N \to \mathbb{R}$. Let us consider a discrete approximation of the gradient, a map $\overline{\nabla} H : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$, satisfying

$$\begin{aligned} H(x) - H(y) &= \overline{\nabla} H(x,y)^{\top}(x-y), \\ \overline{\nabla} H(x,x) &= \nabla H(x), \end{aligned}$$

for any $x, y \in \mathbb{R}^N$. We call $\overline{\nabla} H$ a *discrete gradient*.

Theorem 2.17 (Gonzalez [88])**.** The discrete gradient method

$$\frac{y_{n+1} - y_n}{h} = J^{-1} \overline{\nabla} H(y_{n+1}, y_n)$$

is energy-preserving.

Proof.

$$H(y_{n+1})-H(y_n)=h\overline{\nabla}H(y_{n+1},y_n)^\top\frac{y_{n+1}-y_n}{h}=h\overline{\nabla}H(y_{n+1},y_n)^\top J^{-1}\overline{\nabla}H(y_{n+1},y_n)=0.$$

The first equality is due to the discrete chain rule. The last equality follows from the skew-symmetry of J^{-1}. □

Although the essential idea had been known since 1970s (see, e.g., Greenspan [90]), Gonzalez [88] in 1996 formulated the method explicitly for the first time in the context of numerical analysis (see also McLachlan et al. [141]).

Let us turn to the construction of a discrete gradient. Note that in general a discrete gradient is not unique. Several construction methods have been proposed. Here we show some of them, but the average vector field method is commonly used these days.

Letting $y_{1/2}=(y_0+y_1)/2$, Gonzalez [88] proposed the discrete gradient of the form

$$\overline{\nabla}H(y_0,y_1):=\nabla H(y_{1/2})+\frac{H(y_0)-H(y_1)-\nabla H(y_{1/2})^\top(y_0-y_1)}{\|y_0-y_1\|^2}(y_0-y_1).$$

This definition is theoretically clear, but there is a big drawback that due to $1/\|y_0-y_1\|^2$ in the second term of the right hand side, each component of the discrete gradient contains all components of y_0 and y_1. Therefore, this definition is impractical.

Before Gonzalez's work, Itoh–Abe [116] had considered their approach to construct energy-preserving integrators. For example, when $d=1$ ($y=(q,p)^\top$), their discrete gradient reads

$$\overline{\nabla}H(y_0,y_1):=\left(\frac{H(q_1,p_1)-H(q_0,p_1)}{q_1-q_0},\frac{H(q_0,p_1)-H(q_0,p_0)}{p_1-p_0}\right)^\top.$$

But since their discrete gradient does not have symmetry, the resulting scheme is of order 1. See Ishimori [115] and references therein for symmetrisation of this discrete gradient.

Quispel–McLaren [167] proposed the average vector field (AVF) method:

$$\overline{\nabla}H(y_0,y_1):=\int_0^1\nabla H(\xi y_0+(1-\xi)y_1)\,\mathrm{d}\xi.$$

Advantages of this definition are discussed by Celledoni et al. [48].

Remark 2.5. Since the discrete gradient method is implicit in general, it requires a nonlinear solver such as the simplified Newton method. To reduce the computational cost, a linearisation technique (linearly-implicit method) has been developed in [61, 137, 138], by relaxing the exact energy-preservation.

The idea is briefly illustrated by $\dot q=p$, $\dot p=-q^2$ (i.e., $H(q,p)=\frac{p^2}{2}+\frac{q^3}{3}$). The AVF method gives an energy-preserving integrator

$$\frac{q_1-q_0}{h}=\frac{p_1+p_0}{2},\qquad\frac{p_1-p_0}{h}=-\frac{p_1^2+p_1p_0+p_0^2}{3}.$$

On the other hand, the linearly-implicit method first defines the modified energy

$$\hat H(q_1,q_0,p_1,p_0)=\frac{p_1^2+p_0^2}{4}+\frac{q_1^2q_0+q_1q_0^2}{6},$$

so that it is symmetric in terms of q_1 and q_2 (and similarly p_1 and p_2), and quadratic in terms of q_1 and p_1. Note that the second constraint indicates that if the order of the Hamiltonian is bigger than 4, additional timesteps are required. If we define a multistep scheme by

$$\frac{q_2-q_0}{2h}=\frac{q_2+q_0}{2},\qquad\frac{p_2-p_0}{2h}=-\frac{q_1(q_2+q_1+q_0)}{3},$$

the numerical solution satisfies

$$\hat{H}(q_2, q_1, p_2, p_1) = \hat{H}(q_1, q_0, p_1, p_0).$$

2.5.2 Energy-preserving continuous stage Runge–Kutta methods for Hamiltonian systems

High-order energy-preserving integrators can be constructed by the combination of a symmetric discrete gradient method and the composition method. However, it is not straightforward to derive high-order integrators by collocation-like methods. In 2010, Hairer succeeded in generalising the AVF method to arbitrary high-order [94]. The generalisation was made possible by slightly changing the idea of the collocation method (Definition 2.7).

Definition 2.10 (AVF collocation method [94])**.** Let $c_1, \dots, c_s$ be distinct real numbers ($0 \le c_1 < \cdots < c_s \le 1$). The collocation polynomial $u(t)$ is a polynomial of degree s satisfying

$$\begin{aligned} &u(t_0) = y_0, \\ &\dot{u}(t_0 + c_i h) = \frac{1}{b_i} \int_0^1 l_i(\tau) f(u(t_0 + \tau h))\, \mathrm{d}\tau, \qquad i = 1, \dots, s, \end{aligned}$$

where

$$l_i(\tau) = \prod_{j=1,\ j\neq i}^{s} \frac{\tau - c_j}{c_i - c_j}, \qquad b_i = \int_0^1 l_i(\tau)\, \mathrm{d}\tau,$$

and the numerical solution of the next time step is defined by $y_1 = u(t_0 + h)$.

Theorem 2.18 ([94])**.** The numerical solution of the AVF collocation method satisfies $H(y_1) = H(y_0)$.

Proof will be given in Chapter 3 in a more general context.

If the c_i values are the zeros of the s-th shifted Legendre polynomial, then the method has order $p = 2s$. Here lists concrete expressions of $A_{\tau,\sigma}$ for $s = 1, 2, 3$:

$$\begin{aligned} s = 1: &\quad A_{\tau,\sigma} = \tau, \\ s = 2: &\quad A_{\tau,\sigma} = \tau((4 - 3\tau) - 6(1 - \tau)\sigma), \\ s = 3: &\quad A_{\tau,\sigma} = \tau\big((9 - 18\tau + 10\tau^2) - 12(3 - 8\tau + 5\tau^2)\sigma + 30(1 - 3\tau + 2\tau^2)\sigma^2\big). \end{aligned} \tag{2.15}$$

In Theorem 2.5, we saw that the collocation method can be interpreted as the Runge–Kutta method. But according to Theorem 2.16, the AVF collocation method cannot be interpreted as the standard Runge–Kutta method. Instead, Hairer showed that the AVF collocation method belongs to so called continuous stage Runge–Kutta methods.

Definition 2.11 (Continuous stage Runge–Kutta method)**.** Let $A_{\tau,\sigma}$ and $B_\sigma = A_{1,\sigma}$ be polynomials with respect to the variables in the subscripts. Assume that $A_{0,\sigma} = 0$. We search for a polynomial $Y_\tau \approx u(t_0 + \tau h)$ ($\tau \in [0, 1]$) and $y_1 \approx y(t_0 + h)$ satisfying

$$\begin{aligned} Y_\tau &= y_0 + h \int_0^1 A_{\tau,\sigma} f(Y_\sigma)\, \mathrm{d}\sigma, \\ y_1 &= y_0 + h \int_0^1 B_\tau f(Y_\tau)\, \mathrm{d}\tau. \end{aligned}$$

This one-step method $y_0 \mapsto y_1$ is called a continuous stage Runge–Kutta (CSRK) method.

In the above definition, it is obvious that $y_1 = Y_1$ because of $B_\sigma = A_{1,\sigma}$. Note that in general, we do not have to restrict functions to polynomials, and there is a more general definition (see, e.g., [29, 182]).

It is verified that the AVF collocation method can be interpreted as a CSRK method with the relation

$$A_{\tau,\sigma} = \sum_{i=1}^{s} \frac{1}{b_i} \int_0^\tau l_i(\alpha)\,\mathrm{d}\alpha\, l_i(\sigma), \qquad B_\sigma = 1.$$

Remark 2.6. Brugnano et al. have recently developed the so called Hamiltonian boundary value method (see, e.g., [22, 23]). This method coincides with the AVF method for polynomial Hamiltonian systems. For a specific polynomial Hamiltonian system, an energy-preserving Runge–Kutta method can be derived based on the Hamiltonian boundary value method. However, such a Runge–Kutta method is not energy-preserving for other Hamiltonian systems. Readers should not confuse this with Theorem 2.16.

2.5.3 Conjugate symplecticity

Symplectic methods exactly preserve the symplecticity and nearly preserve the Hamiltonian. Similarly, it is of interest to consider to what extent energy-preserving methods inherit the symplecticity. As a criterion, conjugate symplecticity has been considered recently.

Definition 2.12 (Conjugate symplecticity [97, Section VI.8])**.** A numerical method Φ_h of order p is said to be conjugate symplectic up to order $p + r$ $(r \geq 0)$, if there exists a change of coordinates $z = \chi(y)$ that is $\mathcal{O}(h^p)$ close to the identity such that $\Psi_h = \chi \circ \Phi_h \circ \chi^{-1}$ satisfies

$$\Psi_h'(z)^\top J^{-1} \Psi_h'(z) = J^{-1} + \mathcal{O}(h^{p+r+1}).$$

The method Ψ_h has the same order as Φ_h, and the modified equation is Hamiltonian up to the term h^{p+r-1}.

Conjugate symplecticity of B-series integrators has been extensively studied by Hairer–Zbinden [102]. Below, some important results are summarised without proof.

Theorem 2.19 (Hairer [94, Theorem 7])**.** A symmetric composition of order four based on the average vector field method cannot be conjugate-symplectic up to an higher than four.

Theorem 2.20 (Hairer–Zbinden [102, Theorem 6.3])**.** The AVF collocation method of order $2s$ is conjugate-symplectic up to order $2s + 2$, but it is not conjugate symplectic up to a higher order.

Theorem 2.21 (Hairer–Zbinden [102, Theorem 5.11])**.** A B-series integrator of order $2s$ satisfying the simplifying assumption $C(s)$ and $D(s-1)$ is always conjugate symplectic up to order $2s + 2$. For $s \geq 2$, it is conjugate symplectic up to order $2s + 4$ if and only if

$$\begin{aligned}
&(s+2)(s+1)a(t_s,[\bullet,t_{s+1}]) = (s+1)a(t_s,t_{s+3}) + (s+2)a(t_s,[t_{s+2}]),\\
&(s+2)(s+1)a(t_{s+1},[t_{s+1}]) = (s+2)a(t_{s+1},t_{s+2}) + s(s+2)a(t_s,[t_{s+2}]) - sa(t_s,t_{s+3}).
\end{aligned}$$

Here, t_i denotes a tree with i vertices whose height is 1, e.g., $t_1 = \bullet$, $t_2 =$, $t_3 =$ and $t_4 =$. $a(u,v)$ is defined by

$$a(u,v) := a(u \circ v) + a(v \circ u) - a(u)a(v)$$

where

$$u \circ v = [u_1, \ldots, u_m, v] \qquad \text{for} \qquad u = [u_1, \ldots, u_m].$$

2.5.4 Energy-preserving method based on Euler–Lagrange equation

The energy-preserving methods described above discretise Hamiltonian systems directly. On the other hand, it is possible to construct energy-preserving integrators by discretising the Euler–Lagrange equation. This approach was proposed by Yaguchi [208].

Below, we consider the Hamiltonian system with the Hamiltonian

$$H(q,p) = \frac{1}{2}p^\top p + \frac{1}{2}q^\top A q + U(q),$$

where A is a positive semi-definite matrix. The Hamiltonian system is equivalent to the Euler–Lagrangian equation (2.7) with the Lagrangian

$$L(q,\dot{q}) = \frac{1}{2}\dot{q}^\top \dot{q} - \frac{1}{2}q^\top A q - U(q).$$

The energy-preservation is verified by Noether's theorem. Let $q_h(t) = q(t-h)$. The variation of the action integral yields

$$\begin{aligned}
0 &= \frac{1}{h}\left(\int_h^{T+h} L(q_h,\dot{q}_h)\,\mathrm{d}t - \int_0^T L(q,\dot{q})\,\mathrm{d}t\right)\\
&= \frac{1}{h}\int_T^{T+h} L(q_h,\dot{q}_h)\,\mathrm{d}t - \frac{1}{h}\int_0^h L(q,\dot{q})\,\mathrm{d}t + \frac{1}{h}\int_h^T (L(q_h,\dot{q}_h) - L(q,\dot{q}))\,\mathrm{d}t\\
&\overset{h\to 0}{\to} L|_{t=T} - L|_{t=0} - \int_0^T \left(\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}}\right)\dot{q}\,\mathrm{d}t - \left[\dot{q}\frac{\partial L}{\partial \dot{q}}\right]_0^T\\
&= \left(L - \dot{q}\frac{\partial L}{\partial \dot{q}}\right)\Big|_{t=T} - \left(L - \dot{q}\frac{\partial L}{\partial \dot{q}}\right)\Big|_{t=0} - \int_0^T \left(\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}}\right)\dot{q}\,\mathrm{d}t.
\end{aligned} \tag{2.16}$$

Therefore, for the Euler–Lagrange equation $\frac{\partial L}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}} = 0$, it follows

$$L - \dot{q}\frac{\partial L}{\partial \dot{q}} = \text{const.},$$

which indicates the preservation of Hamiltonian because $L - \dot{q}\frac{\partial L}{\partial \dot{q}} = -H$.

The idea of deriving an energy-preserving integrator is to mimic the calculation (2.16) in the discrete setting. We first discretise the Lagrangian by

$$L_\mathrm{d}(q_n, \delta_t^+ q_n) := \frac{1}{2}(\delta_t^+ q_n)^\top (\delta_t^+ q_n) - \frac{1}{2}q_n^\top A q_n - U(q_n).$$

Below, we use the abbreviation $L(q_n)$. It then follows that

$$\begin{aligned}
0 &= \frac{1}{h}\left(\sum_{n=1}^{N+1} L_\mathrm{d}(q_{n-1}) - \sum_{n=0}^{N} L_\mathrm{d}(q_n)\right)h\\
&= L_\mathrm{d}(q_N) - L_\mathrm{d}(q_0) + \frac{1}{h}\sum_{n=1}^{N}(L_\mathrm{d}(q_{n-1}) - L_\mathrm{d}(q_n))h\\
&= L_\mathrm{d}(q_N) - L_\mathrm{d}(q_0) - h\sum_{n=1}^{N}\Big((\delta_t^+ q_{n-1/2})^\top(\delta_t^+\delta_t^- q_n) - q_{n-1/2}^\top A(\delta_t^- q_n) - \overline{\nabla}U(q_n,q_{n-1})^\top(\delta_t^- q_n)\Big)\\
&= \big(L_\mathrm{d}(q_N) - (\delta_t^+ q_{N-1/2})^\top(\delta_t^+ q_N)\big) - \big(L_\mathrm{d}(q_0) - (\delta_t^+ q_{-1/2})^\top(\delta_t^+ q_0)\big)\\
&\quad + h\sum_{n=1}^{N}\big(\delta_t^-\delta_t^+ q_{n-1/2} + A q_{n-1/2} + \overline{\nabla}U(q_n,q_{n-1})\big)^\top(\delta_t^- q_n),
\end{aligned}$$

where $\overline{\nabla}U(q_n, q_{n-1})$ denotes the discrete gradient of U. This calculation immediately suggests the numerical scheme

$$\delta_t^- \delta_t^+ q_{n-1/2} + A q_{n-1/2} + \overline{\nabla} U(q_n, q_{n-1}) = 0. \tag{2.17}$$

Theorem 2.22. The scheme (2.17) is energy-preserving in the sense that

$$L_{\mathrm{d}}(q_n) - (\delta_t^+ q_{n-1/2})^\top (\delta_t^+ q_n) = -\left(\frac{1}{2} (\delta_t^- q_n)^\top (\delta_t^+ q_n) + \frac{1}{2} q_n^\top A q_n + U(q_n) \right) = \text{const.}$$

The most remarkable feature of the scheme (2.17) is that it is explicit, though q_1 and q_2 should be given/calculated in advance. However, it should also be noted that the scheme does not preserve the genuine Hamiltonian. It is still open if the scheme is conjugate symplectic.

2.5.5 Energy-preserving partitioned continuous stage Runge–Kutta methods for Poisson systems

In the above subsections, we saw the energy-preserving methods for Hamiltonian systems. In this subsection, we summarise how such methods are extended to Poisson systems. While the projection methods and methods on local coordinates are still applicable, it turns out that a CSRK method cannot be energy-preserving for Poisson systems. The difficulty is due to the dependence of the matrix Λ on $y(t)$. We here review the method by Cohen–Hairer [53], where a partitioned version of CSRK methods, called a partitioned CSRK (PCSRK) method, is considered.

First, let us consider the simplest case, i.e., the discrete gradient method of order two. In the proof of Theorem 2.17, the last equality is due to the skew-symmetry of J^{-1}. Therefore, the matrix $\Lambda(y)$ of Poisson systems should be discretised with y_n and y_{n+1} so that the discrete version is still skew-symmetric. Although such a discretisation is arbitrary, the following way is preferable because it is symmetric

$$\frac{y_{n+1} - y_n}{h} = \Lambda\left(\frac{y_n + y_{n+1}}{2} \right) \overline{\nabla} H(y_{n+1}, y_n).$$

The above example indicates that the matrix $\Lambda(y)$ and gradient $\nabla H(y)$ should be discretised in a different manner. Therefore, the CSRK framework is insufficient for the construction of high-order energy-preserving integrators. Following Cohen–Hairer [53], we here introduce PCSRK methods.

Definition 2.13 (Partitioned continuous stage Runge–Kutta method)**.** Let $A_{i\tau,j\sigma}$ be a polynomial of degree s with respect to the variables τ and σ. We search for a polynomial $Y_\tau \approx u(t_0 + \tau h)$ $(\tau \in [0,1])$, Z_i $(i = 1, \dots, s)$ and $y_1, z_1 \approx y(t_0 + h)$ such that they satisfy

$$\begin{aligned}
Y_\tau &= y_0 + h \sum_{j=1}^{s} \int_0^1 A_{i\tau,j\sigma} \Lambda(Z_j) \nabla H(Y_\sigma) \, \mathrm{d}\sigma, \\
Z_i &= z_0 + h \sum_{j=1}^{s} \int_0^1 \widehat{A}_{i\tau,j\sigma} \Lambda(Z_j) \nabla H(Y_\sigma) \, \mathrm{d}\sigma, \qquad i = 1, \dots, s, \\
y_1 &= y_0 + h \sum_{i=1}^{s} \int_0^1 B_{i\tau} \Lambda(Z_i) \nabla H(Y_\tau) \, \mathrm{d}\tau, \\
z_1 &= z_0 + h \sum_{i=1}^{s} \int_0^1 \widehat{B}_{i\tau} \Lambda(Z_i) \nabla H(Y_\tau) \, \mathrm{d}\tau,
\end{aligned} \tag{2.18}$$

with $y_0 = z_0$, where

- Y_τ is a polynomial in τ of degree s and satisfies $Y_0 = y_0$,
- $0 \le c_1 < \cdots < c_s \le 1$,
- $\widehat{A}_{i\tau,j\sigma} = A_{c_i,j\sigma}$,
- $B_{j\sigma} = \widehat{B}_{j\sigma} = A_{1,j\sigma}$.

This one-step method $y_0 \mapsto y_1$ is called a partitioned continuous stage Runge–Kutta (PCSRK) method.

The notation $A_{i\tau,j\sigma}$, which was introduced in [53], depends on $\tau,\ \sigma \in [0,1]$, $j = 1,\dots,s$ and i. In reality, it does not depend on i, but we leave it as it is because it becomes useful when considering order conditions. In other places, we can simply understand this as $A_{i\tau,j\sigma} = A_{\tau,j\sigma}$.

It is clear that $y_1 = z_1$ and (2.18) is equivalent to $Z_i = Y_{c_i}$. As mentioned in [53], these methods are consistent with the partitioned system of differential equations

$$\begin{aligned} \dot{y} &= \Lambda(z)\nabla H(y), \quad y(t_0) = y_0, \\ \dot{z} &= \Lambda(z)\nabla H(y), \quad z(t_0) = z_0, \end{aligned}$$

whose solutions satisfy $y(t) = z(t)$ if $y_0 = z_0$.

It is shown in [53] that the PCSRK method is energy-preserving independently of $c_1,\dots,c_s$ if

$$A_{i\tau,j\sigma} = \frac{l_j(\sigma)}{b_j} \int_0^\tau l_j(\alpha)\,\mathrm{d}\alpha.$$

Moreover, if $c_1,\dots,c_s$ are the zeros of the sth shifted Legendre polynomial, the method has the accuracy order $2s$.

2.6 Motivation and summary of the subsequent chapters

Chapter 3

For ordinary differential equations with periodic or oscillatory solutions, there have been many branches of research activities. For example, trigonometric methods for second-order ODEs and exponentially-fitted (EF) methods for first-order ODEs have been studied in the last few decades. The trigonometric methods have been mainly developed in the context of highly oscillatory differential equations (see, e.g., [54, 68, 91, 95, 104, 183] and references therein), after the first theoretical foundation given by Gautschi [86] and Lyche [131]. The EF methods were first considered by Simos [175] and Paternoster [162] independently, and then have been developed by several authors [76, 77, 78, 160, 161, 186, 187, 188, 190].

In Chapter 3, we first focus on the EF methods. Recently, symplectic EF methods have been developed for Hamiltonian systems (see, e.g., [36, 37, 38, 39, 40]) by combining the ideas of symplectic methods and EF methods. However, if we turn our attention to energy-preserving integrators, only a few papers (e.g., [199]) have been written in this context, and thus it seems much is left to be investigated. Taking these facts into account, we aim to construct energy-preserving EF methods. Below, the difficulty of this challenge is clarified. The symplectic EF methods have been constructed by the combination of

- the characterisation of Runge–Kutta methods being symplectic (Theorem 2.13), and
- a standard theory of exponentially-fitted Runge–Kutta methods (which will be summarised in Chapter 3).

As we saw in the previous section, Runge–Kutta methods cannot be energy-preserving, and thus we are forced to construct our intended methods in a framework of CSRK methods. Actually we show that we can newly develop

- characterisation of CSRK methods being energy-preserving,
- a standard theory of exponentially-fitted CSRK methods

and combine the two theories to derive energy-preserving EF integrators. We would like to emphasise that the energy-preservation characterisation, which corresponds to Theorem 2.13, is an important theorem in a more general context of geometric numerical integration methods.

Next, we extend the above ideas to Poisson systems.

Finally, we briefly consider the trigonometric methods. It is known that several existing trigonometric methods are symplectic. We show that energy-preserving trigonometric methods can also be constructed by using the theory summarised in Section 2.5.4.

Chapter 4

Butcher–Imran [33] have recently shown that we can construct efficient high-order symplectic integrators for Hamiltonian systems. In fact they constructed fourth-order symplectic integrators, whose computational costs are comparable to the midpoint rule (the simplest second-order symplectic method) if parallelism is available. Motivated by their work, in this chapter, we consider a derivation of efficient, high-order energy-preserving methods for Hamiltonian systems. The accuracy order can be easily increased by the composition method (Section 2.2.3) based on the AVF method. However, there is a tradeoff between the accuracy order and computational cost. For example, if we employ the triple-jump technique (2.6), we have to call an AVF integrator 3^{s-1} times per each timestep in order to achieve $2s$-order. The AVF collocation method (Definition 2.10) is an alternative way to achieve $2s$-order. In this method, we have to solve a system of nonlinear equations of size sN once per each timestep.

The aim of Chapter 4 is to derive a new family of energy-preserving methods which can be high-order, and at the same time, whose computational cost is just comparable to that of the AVF method when parallelism is available. We here would like to emphasise that in this thesis we do not intend to implement the existing high-order methods in parallel architecture by making the best use of characteristics of each problem, though such a technique can also be incorporated with the proposed method.

Our idea is summarised as follows. Our derivation is based on continuous stage Runge–Kutta (CSRK) methods. For conventional RK methods, an s-stage implicit RK method can be implemented in parallel with s processors, if the coefficient matrix has only real, distinct eigenvalues. Precisely speaking, in such cases, if we apply the simplified Newton method, the resulting linear system of size sN can be divided into s linear systems of size N. Motivated by this fact, we shall seek a similar condition for CSRK methods. We then derive new CSRK integrators satisfying this condition, characterisations for energy-preservation, and order conditions. Note that the characterisation of CSRK methods being energy-preserving is given in Section 3.2. Also note that to check the order conditions is the most cumbersome part in the derivation, and in order to avoid such a heavy task, we characterise the order conditions in terms of the coefficient polynomial of CSRK methods.

Chapter 3

Energy-preserving exponentially-fitted/trigonometric integrators for Hamiltonian/Poisson systems

In this chapter, we consider energy-preserving numerical methods for differential equations with periodic or oscillatory solutions. Section 3.1 briefly reviews the basic concepts of exponentially-fitted Runge–Kutta (EFRK) methods and symplectic EFRK methods. A characterisation of CSRK methods being energy-preserving, and theory of EF continuous stage Runge–Kutta (CSRK) methods are shown in Sections 3.2 and 3.3, respectively. Energy-preserving EF methods for Hamiltonian systems are then constructed in Section 3.4, and they are further extended to Poisson systems in Section 3.5. Energy-preserving trigonometric methods are considered in Section 3.6.

In this chapter we use several abbreviations. The following table shows their list.

RK	Runge–Kutta
EF	exponentially-fitting, exponentially-fitted
FF	functionally-fitting, functionally-fitted
EF(FF)RK	exponentially-fitted (functionally-fitted) Runge–Kutta
SEFRK	symplectic EFRK
(P)CSRK	partitioned continuous stage Runge–Kutta
EPCSRK	energy-preserving CSRK
EFCSRK	exponentially-fitted CSRK
EPEFCSRK	energy-preserving exponentially-fitted CSRK

3.1 A brief review of exponentially-fitted Runge–Kutta methods and symplectic exponentially-fitted Runge–Kutta methods

In this section, we briefly review the basic concepts of EFRK methods and symplectic EFRK (SEFRK) methods.

3.1.1 Characterisations of symplecticity and symmetry of modified RK methods

We already saw the characterisations for symplecticity and symmetry in terms of RK methods in Chapter 2 (Theorem 2.13). Just for the explanation of symplectic EF methods, we introduce modified RK methods.

We consider an s-stage modified Runge–Kutta (mRK) method defined by

$$Y_i = \gamma_i y_0 + h\sum_{j=1}^{s} a_{ij} f(Y_j), \qquad i = 1,\dots,s, \tag{3.1}$$

$$y_1 = y_0 + h\sum_{i=1}^{s} b_i f(Y_i), \tag{3.2}$$

where $y_1 \approx y(t_0+h)$, $Y_i \approx y(t_0+c_i h)$ $(i=1,\dots,s)$ and the real parameters c_i and b_i $(i=1,\dots,s)$ denote the nodes and the weights of the method. In the standard RK methods, all $\gamma_i = 1$, but several authors introduced the γ_i parameters in the context of EF methods [76, 186, 187]. We often refer to (3.1) and (3.2) as the internal and final stages, respectively. The mRK method (3.1) and (3.2) is often represented by means of Butcher's tableau

$$\begin{array}{c|c|c} c & \gamma & A \\ \hline & & b^\top \end{array} = \begin{array}{c|c|ccc} c_1 & \gamma_1 & a_{11} & \cdots & a_{1s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & \gamma_s & a_{s1} & \cdots & a_{ss} \\ \hline & & b_1 & \cdots & b_s \end{array}$$

or equivalently the quartet (c,γ,A,b).

As an extension of Theorem 2.13, the characterisation of mRK methods being symplectic was obtained by Van de Vyver [184, 185].

Theorem 3.1 (Van de Vyver [184, 185])**.** A mRK method solving Hamiltonian systems is symplectic if the following conditions are satisfied

$$b_j\frac{a_{ji}}{\gamma_j} + b_i\frac{a_{ij}}{\gamma_i} - b_i b_j = 0, \qquad 1 \le i,j \le s.$$

In [36], for mRK methods whose coefficients are even functions of h, the symmetry conditions are given by

$$c + Sc = e, \qquad b = Sb, \qquad \gamma = S\gamma, \qquad SA + AS = \gamma b^\top, \tag{3.3}$$

where

$$e = (1,\dots,1)^\top \in \mathbb{R}^s \qquad \text{and} \qquad S = (s_{ij}) \in \mathbb{R}^{s\times s} \qquad \text{with} \qquad s_{ij} = \begin{cases} 1, & \text{if } i+j = s+1, \\ 0, & \text{otherwise.} \end{cases}$$

3.1.2 Exponentially-fitted RK methods

In this subsection, we briefly review the basic concepts of EFRK methods. In the context of RK methods, a formula can be constructed by the collocation approach: we choose the available parameters (c,γ,A,b) so that the resulting scheme exactly solves problems whose solution belongs to the linear space spanned by

$$\mathcal{F} = \{u_1(t), u_2(t), \dots, u_r(t)\}, \quad r \le s.$$

The set $\mathcal{F} = \{1, t, t^2, \dots, t^s\}$ is usually considered. In this case, the parameters of the resulting scheme are independent of h. If $\mathcal{F}$ contains exponential or trigonometric functions, these methods are called EFRK methods. In more general cases in which $\mathcal{F}$ contains general functions, these methods are called functionally-fitted RK (FFRK) methods [160, 161]. In general, the coefficients (c,γ,A,b) of an EFRK

or FFRK method may depend on not only the fitting functions $u_1,\dots,u_r$, but also the step size h. The coefficients of the FFRK method (3.1)–(3.2) are determined by the linear systems

$$u_k(t_0e+hc)-\gamma u_k(t_0)=hAu_k'(t_0e+hc),\quad k=1,\dots,r, \tag{3.4}$$

$$u_k(t_0+h)-u_k(t_0)=hb^\top u_k'(t_0e+hc),\qquad k=1,\dots,r, \tag{3.5}$$

where $e=(1,\dots,1)^\top\in\mathbb{R}^s$ and we use the notation $g(v)=(g(v_1),\dots,g(v_s))^\top$ for $v=(v_1,\dots,v_s)\in\mathbb{R}^s$ and a scalar function g.

In general cases, the coefficients may depend on t_0, h and $\mathcal{F}$, but under some standard requirements on $\mathcal{F}$, they are independent of t_0.

Let us consider the solvability of the systems (3.4) and (3.5). When $r=s$, the coefficients b and A are uniquely determined for all $h>0$ and $t\in[t_0,T]$, if the matrix

$$M(t,h)=\begin{pmatrix} u_1'(t+c_1h) & \cdots & u_1'(t+c_sh)\\ \vdots & \ddots & \vdots\\ u_s'(t+c_1h) & \cdots & u_s'(t+c_sh)\end{pmatrix}$$

is non-singular [160]. Below we explain the key idea of the proof of this statement. If the functions $u_k(t)$ $(k=1,\dots,s)$ are sufficiently smooth, from the Taylor expansion we have

$$M(t,h)=W^\top(t)\begin{pmatrix} 1 & 1 & \cdots & 1\\ c_1h & c_2h & \cdots & c_sh\\ \vdots & \vdots & \ddots & \vdots\\ \frac{(c_1h)^{s-1}}{(s-1)!} & \frac{(c_2h)^{s-1}}{(s-1)!} & \cdots & \frac{(c_sh)^{s-1}}{(s-1)!}\end{pmatrix}+\mathcal{O}(h^s), \tag{3.6}$$

where $W(t)$ is the Wronskian matrix defined by

$$W(t):=\begin{pmatrix} u_1'(t) & \cdots & u_s'(t)\\ \vdots & \ddots & \vdots\\ u_1^{(s)}(t) & \cdots & u_s^{(s)}(t)\end{pmatrix}\quad\text{where}\quad u^{(i)}:=\frac{\mathrm{d}^i}{\mathrm{d}t^i}u(t).$$

Therefore, due to the continuity of determinant, if the nodes are distinct ($c_i\neq c_j$, $i\neq j$) and $W(t)$ is non-singular, the coefficients b and A are uniquely determined.

In the context of EFRK methods, we usually consider

$$\mathcal{F}_1=\{\exp(\lambda t),\exp(-\lambda t)\} \tag{3.7}$$

or $\mathcal{F}_2=\{\cos(\omega t),\sin(\omega t)\}$. Note that $\mathcal{F}_2$ is obtained from $\mathcal{F}_1$ with $\lambda=i\omega$. When we consider the set $\mathcal{F}_1$, the linear systems (3.4)–(3.5) reduce to

$$A\cosh(cz)=\frac{\sinh(cz)}{z},\quad A\sinh(cz)=\frac{\cosh(cz)-\gamma}{z}, \tag{3.8}$$

$$b^\top\cosh(cz)=\frac{\sinh(z)}{z},\quad b^\top\sinh(cz)=\frac{\cosh(z)-1}{z}, \tag{3.9}$$

where $z=\lambda h$. For $s=2$, by the above statement, the coefficients b and A are uniquely determined in terms of the nodes c_i and parameters γ_i. By simply choosing the Gaussian nodes $(c_1,c_2)=(\frac{1}{2}-\frac{\sqrt{3}}{6},\frac{1}{2}+\frac{\sqrt{3}}{6})$ and $\gamma_1=\gamma_2=1$, we can obtain the fourth order EFRK method which reduces to the two-stage Gauss method when $\lambda=0$ [190]. Unfortunately, however, this EF method is not symplectic as shown in [40], which indicates that the derivation of symplectic EFRK methods needs some more tricks.

Table 3.1: Symplectic EFRK methods.

2nd order	Van de Vyver [184]
4th order	Vyver [185], Calvo et al. [37]
6th order	Calvo et al. [36, 38]
8th order	Calvo et al. [40], Vanden Berghe–Van Daele [189]
2s order	Calvo et al. [39]

3.1.3 Symplectic exponentially-fitted RK methods

Recently, some symplectic (and symmetric) EFRK (SEFRK) methods have been proposed by several authors. In Table 3.1, some existing methods are shown.

We illustrate here the key for the construction of SEFRK methods, following [37]. By taking the derivation of fourth order SEFRK scheme as our example, let us start with the two stage mRK formulation

$$\begin{array}{c|c|cc} c_1 & \gamma_1 & a_{11} & a_{12} \\ c_2 & \gamma_2 & a_{21} & a_{22} \\ \hline & & b_1 & b_2 \end{array}.$$

The coefficients should be related by

$$\begin{array}{c|c|cc} c_1 & \gamma_1 & a_{11} & a_{12} \\ c_2 & \gamma_2 & a_{21} & a_{22} \\ \hline & & b_1 & b_2 \end{array} = \begin{array}{c|c|cc} \frac{1}{2}-\theta & \gamma & \frac{\gamma b}{2} & a \\ \frac{1}{2}+\theta & \gamma & \gamma b - a & \frac{\gamma b}{2} \\ \hline & & b & b \end{array} \tag{3.10}$$

so that the method is symplectic and symmetric. Here, Theorem 3.1 and (3.3) were used.

When we consider the set of functions $\mathcal{F}_1$ (3.7), the linear systems (3.4)–(3.5) reduce to (3.8)–(3.9). Firstly, from $b_1 = b_2 = b$, $c_{1,2} = \frac{1}{2} \mp \theta$ and (3.9), we can easily obtain

$$b_1 = b_2 = b = \frac{\sinh(\frac{z}{2})}{z\cosh(\theta z)}.$$

Next, from $\gamma_1 = \gamma_2 = \gamma$ and (3.8), the coefficients A are given by

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \frac{1}{z\sinh(2\theta z)} \begin{pmatrix} \gamma\cosh((\frac{1}{2}+\theta)z) - \cosh(2\theta z) & 1-\gamma\cosh((\frac{1}{2}-\theta)z) \\ -1+\gamma\cosh((\frac{1}{2}+\theta)z) & -\gamma\cosh((\frac{1}{2}-\theta)z) + \cosh(2\theta z) \end{pmatrix}.$$

The parameter γ can be also determined as

$$\gamma_1 = \gamma_2 = \gamma = \frac{2\cosh(2\theta z)}{\cosh((\frac{1}{2}+\theta)z) + \cosh((\frac{1}{2}-\theta)z)}$$

in order to satisfy the relation (3.10). Finally we select the parameter θ using the order conditions. The resulting scheme is of order four if $\theta = \sqrt{3}/6$. This fourth order SEFRK scheme coincides with that obtained in [185].

Remark 3.1. As pointed out in almost all papers dealing with EF methods, the coefficients are subject to heavy cancellation when evaluated for small values of $|z|$, because the denominator gets close to 0.

In that case the following series expansions should be used:

$$
\begin{aligned}
a_{11} &= a_{22} = \frac{1}{4} - \frac{7}{8640}z^4 + \frac{31}{272160}z^6 - \frac{167}{13063680}z^8 + \cdots, \\
a_{12} &= \frac{1}{4} - \frac{\sqrt{3}}{6} + \frac{\sqrt{3}}{216}z^2 - \left(\frac{7}{8640} + \frac{\sqrt{3}}{6480}\right)z^4 + \left(\frac{31}{272160} + \frac{17\sqrt{3}}{3265920}\right)z^6 \\
&\quad - \left(\frac{167}{13063680} + \frac{31\sqrt{3}}{176359680}\right)z^8 + \cdots, \\
a_{21} &= \frac{1}{4} + \frac{\sqrt{3}}{6} - \frac{\sqrt{3}}{216}z^2 + \left(-\frac{7}{8640} + \frac{\sqrt{3}}{6480}\right)z^4 + \left(\frac{31}{272160} - \frac{17\sqrt{3}}{3265920}\right)z^6 \\
&\quad + \left(-\frac{167}{13063680} + \frac{31\sqrt{3}}{176359680}\right)z^8 + \cdots, \\
\gamma &= 1 - \frac{1}{288}z^4 + \frac{1}{2160}z^6 - \frac{881}{17418240}z^8 + \cdots \\
b &= \frac{1}{2} + \frac{1}{8640}z^4 - \frac{1}{272160}z^6 + \frac{13}{104509440}z^8 + \cdots.
\end{aligned}
$$

It is clear that in the limit $z \to 0$ the well known classical fourth-order Gauss method is recovered.

3.2 Characterisations of energy-preservation and symmetry

In this section, we present characterisations of CSRK methods being energy-preserving and symmetric for Hamiltonian systems. We also show similar characterisations of partitioned CSRK (PCSRK) methods being energy-preserving and symmetric for Poisson systems.

3.2.1 CSRK methods and their characterisations of energy-preservation and symmetry for Hamiltonian systems

As defined in Definition 2.11, we consider an s-degree CSRK method defined by

$$Y_\tau = y_0 + h\int_0^1 A_{\tau,\sigma} f(Y_\sigma)\,\mathrm{d}\sigma, \tag{3.11}$$

$$y_1 = y_0 + h\int_0^1 B_\sigma f(Y_\tau)\,\mathrm{d}\tau, \tag{3.12}$$

where Y_τ is a polynomials of degree s with respect to τ satisfying $Y_0 = y_0$, and $A_{\tau,\sigma}$ and B_σ are polynomial with respect to the variables in the subscripts. The following theorem show a characterisation of CSRK methods being energy-preserving.

Theorem 3.2. A CSRK method solving Hamiltonian systems is energy-preserving if $\frac{\partial}{\partial\tau}A_{\tau,\sigma}$ is symmetric, i.e.,

$$A'_{\tau,\sigma} = A'_{\sigma,\tau} \quad \text{where} \quad A'_{\tau,\sigma} := \frac{\partial}{\partial\tau}A_{\tau,\sigma}. \tag{3.13}$$

Proof. We can express $\frac{\partial}{\partial\tau}A_{\tau,\sigma}$ as

$$\frac{\partial}{\partial\tau}A_{\tau,\sigma} = \sum_{l=0}^{s-1} a'(l,l)\tau^l\sigma^l + \sum_{m<n}\big(a'(m,n)\tau^m\sigma^n + a'(n,m)\tau^n\sigma^m\big).$$

Note that the symmetry of $\frac{\partial}{\partial\tau}A_{\tau,\sigma}$ is equivalent to $a'(m,n)=a'(n,m)$. Thus we have

$$\begin{aligned}
&H(y_1)-H(y_0)=\int_0^1\frac{\mathrm{d}}{\mathrm{d}\tau}H(Y_\tau)\,\mathrm{d}\tau=\int_0^1\dot{Y}_\tau{}^\top\nabla H(Y_\tau)\,\mathrm{d}\tau\\
&=h\int_0^1\left(\int_0^1\frac{\partial}{\partial\tau}A_{\tau,\sigma}J^{-1}\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top\nabla H(Y_\tau)\,\mathrm{d}\tau\\
&=h\sum_{l=0}^{s-1}a'(l,l)\left(\int_0^1\sigma^l\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top J^{-\top}\int_0^1\tau^l\nabla H(Y_\tau)\,\mathrm{d}\tau\\
&\quad+h\sum_{m<n}\left\{a'(m,n)\left(\int_0^1\sigma^n\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top J^{-\top}\int_0^1\tau^m\nabla H(Y_\tau)\,\mathrm{d}\tau\right.\\
&\qquad\left.+a'(n,m)\left(\int_0^1\sigma^m\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top J^{-\top}\int_0^1\tau^n\nabla H(Y_\tau)\,\mathrm{d}\tau\right\}=0.
\end{aligned}$$

In the last equality, the first term vanishes due to the skew-symmetry of J. The second term vanishes because of

$$\begin{aligned}
&\left(\int_0^1\sigma^n\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top J^{-\top}\int_0^1\tau^m\nabla H(Y_\tau)\,\mathrm{d}\tau\\
&=-\left(\int_0^1\sigma^m\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top J^{-\top}\int_0^1\tau^n\nabla H(Y_\tau)\,\mathrm{d}\tau
\end{aligned}$$

and the symmetry $a'(m,n)=a'(n,m)$. □

The symmetry condition of a CSRK method is shown in [94].

Theorem 3.3 ([94]). A CSRK method is symmetric if

$$A_{1-\tau,1-\sigma}+A_{\tau,\sigma}=B_\sigma. \tag{3.14}$$

This statement is still true even when the coefficients are even functions of h.

3.2.2 PCSRK methods and their characterisations of energy-preservation and symmetry for Poisson systems

As defined in Definition 2.13, we consider an s-degree PCSRK method defined by

$$\begin{aligned}
Y_\tau&=y_0+h\sum_{j=1}^s\int_0^1A_{i\tau,j\sigma}\Lambda(Z_j)\nabla H(Y_\sigma)\,\mathrm{d}\sigma,\\
Z_i&=z_0+h\sum_{j=1}^s\int_0^1\widehat{A}_{i\tau,j\sigma}\Lambda(Z_j)\nabla H(Y_\sigma)\,\mathrm{d}\sigma\quad(i=1,\dots,s),\\
y_1&=y_0+h\sum_{i=1}^s\int_0^1B_{i\tau}\Lambda(Z_i)\nabla H(Y_\tau)\,\mathrm{d}\tau,\\
z_1&=z_0+h\sum_{i=1}^s\int_0^1\widehat{B}_{i\tau}\Lambda(Z_i)\nabla H(Y_\tau)\,\mathrm{d}\tau,
\end{aligned}$$

with $y_0=z_0$, where

- Y_τ is a polynomial in τ of degree s and satisfies $Y_0 = y_0$,
- $A_{i\tau,j\sigma}$ is a polynomial in τ and σ,
- $0 \le c_1 < \cdots < c_s \le 1$,
- $\widehat{A}_{i\tau,j\sigma} = A_{c_i,j\sigma}$,
- $B_{j\sigma} = \widehat{B}_{j\sigma} = A_{1,j\sigma}$.

Theorem 3.4. A PCSRK method solving Poisson systems is energy-preserving if $\frac{\partial}{\partial\tau}A_{i\tau,j\sigma}$ is symmetric for all $j = 1,\dots,s$.

Proof. We can express each $\frac{\partial}{\partial\tau}A_{i\tau,j\sigma}$ as

$$\frac{\partial}{\partial\tau}A_{i\tau,j\sigma} = \sum_{l=0}^{s-1} a_j(l,l)\tau^l\sigma^l + \sum_{m<n}\left(a_j(m,n)\tau^m\sigma^n + a_j(n,m)\tau^n\sigma^m\right).$$

Note that the symmetry of $\frac{\partial}{\partial\tau}A_{i\tau,j\sigma}$ is equivalent to $a_j(m,n) = a_j(n,m)$ for all j,m,n. Thus we have

$$\begin{aligned}
H(y_1) - H(y_0) &= \int_0^1 \frac{\mathrm{d}}{\mathrm{d}\tau}H(Y_\tau)\,\mathrm{d}\tau = \int_0^1 \dot{Y}_\tau^\top \nabla H(Y_\tau)\,\mathrm{d}\tau \\
&= h\int_0^1 \left(\sum_{j=1}^{s}\int_0^1 \frac{\partial}{\partial\tau}A_{i\tau,j\sigma}\Lambda(Z_j)\nabla H(Y_\tau)\,\mathrm{d}\sigma\right)^\top \nabla H(Y_\tau)\,\mathrm{d}\tau \\
&= h\sum_{j=1}^{s}\sum_{l=0}^{s-1} a_j(l,l)\left(\int_0^1 \sigma^l\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top \Lambda^\top(Z_j)\int_0^1 \tau^l\nabla H(Y_\tau)\,\mathrm{d}\tau \\
&\quad + h\sum_{j=1}^{s}\sum_{m<n}\left\{a_j(m,n)\left(\int_0^1 \sigma^n\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top \Lambda^\top(Z_j)\int_0^1 \tau^m\nabla H(Y_\tau)\,\mathrm{d}\tau\right. \\
&\quad \left. + a_j(n,m)\left(\int_0^1 \sigma^m\nabla H(Y_\sigma)\,\mathrm{d}\sigma\right)^\top \Lambda^\top(Z_j)\int_0^1 \tau^n\nabla H(Y_\tau)\,\mathrm{d}\tau\right\} = 0.
\end{aligned}$$

□

Symmetry condition is also given as follows.

Theorem 3.5. A PCSRK method is symmetric if $A_{i(1-\tau),(s+1-j)(1-\sigma)} + A_{i\tau,j\sigma} = A_{1,j\sigma}$, and the nodes c_i are symmetric, i.e., $c_{s+1-i} = 1 - c_i$.

Proof. Based on Theorem 2.8, it is checked that the PCSRK method is symmetric if

$$\widehat{A}_{(s+1-i)(1-\tau),(s+1-j)(1-\sigma)} + \widehat{A}_{s\tau,j\sigma} = \widehat{B}_{j\sigma}.$$

This condition holds if $A_{i(1-\tau),(s+1-j)(1-\sigma)} + A_{i\tau,j\sigma} = A_{1,j\sigma}$ and $c_{s+1-i} = 1 - c_i$ are satisfied. □

3.3 Exponentially-fitted CSRK methods

In this section, we develop a theory of standard EFCSRK methods. But we consider a wider class of functionally-fitted CSRK (FFCSRK) methods which contain EFCSRK methods as special cases. For s-stage FFRK methods, we can consider the fitting on $s+1$ nodes, i.e., $t = t_0 + c_1 h, \dots, t_0 + c_s h, t_0 + h$, because the coefficients A and b can be chosen independently. However, for s-degree FFCSRK methods, since $A_{\tau,\sigma}$ and B_σ are dependent, we can consider the fitting on only s nodes, and one of them should be $t = t_0 + h$. This fact is the biggest difference between FFRK methods and FFCSRK methods.

The coefficients $A_{\tau,\sigma}$ and B_σ of a FFCSRK method are determined by the linear systems

$$u_k(t_0 e + hc) - u_k(t_0) = h \int_0^1 A_{c,\sigma} \widetilde{u_k'}(t_0 + \sigma h)\, \mathrm{d}\sigma, \quad k = 1, \dots, r, \tag{3.15}$$

$$u_k(t_0 e + h) - u_k(t_0) = h \int_0^1 B_\sigma \widetilde{u_k'}(t_0 + \sigma h)\, \mathrm{d}\sigma, \qquad k = 1, \dots, r, \tag{3.16}$$

where $c = (c_1, \dots c_{s-1})^\top \in \mathbb{R}^{s-1}$, $e = (1, \dots, 1)^\top \in \mathbb{R}^{s-1}$. Here we also introduced the following notation: for a scalar function g and a vector $v = (v_1, \dots, v_{s-1})^\top \in \mathbb{R}^{s-1}$, $g(v)$ means the abbreviation $g(v) = (g(v_1), \dots, g(v_{s-1}))^\top$, and $\widetilde{g}(t_0 + \sigma h)$ denotes a polynomial of degree s which is a linear combination of $g(t_0), g(t_0 + c_1 h), \dots, g(t_0 + c_{s-1} h), g(t_0 + h)$.

Proposition 3.1. Assume that $u_1, \dots, u_r$ are sufficiently smooth. When $r = s$, the coefficients $A_{\tau,\sigma}$ are uniquely determined for all h and $t \in [t_0, T]$ if the Wronskian matrix

$$W(t) := \begin{pmatrix} u_1'(t) & \cdots & u_s'(t) \\ \vdots & \ddots & \vdots \\ u_1^{(s)}(t) & \cdots & u_s^{(s)}(t) \end{pmatrix}$$

is non-singular and the nodes $0 < c_1, \dots, c_{s-1} < 1$ are distinct.

Proof. Since $A_{c,\sigma}$ and $\widetilde{u_k'}(t_0 + \sigma h)$ are polynomials of degree $s-1$ and s in terms of σ, the right hand side of (3.15) can be integrated exactly by the s-points Gaussian quadrature rule:

$$h \int_0^1 A_{c_i,\sigma} \widetilde{u_k'}(t_0 + \sigma h)\, \mathrm{d}\sigma = h \sum_{j=1}^{s} b_j' A_{c_i,c_j'} \widetilde{u_k'}(t_0 + c_j' h)$$

where c_j' $(j = 1, \dots, s)$ denote the Gaussian nodes and

$$b_i' = \int_0^1 \prod_{j=1, j\neq i}^{s} \frac{\tau - c_j'}{c_i' - c_j'}\, \mathrm{d}\tau.$$

Therefore if the matrix

$$\begin{pmatrix} \widetilde{u_1'}(t + c_1' h) & \cdots & \widetilde{u_1'}(t + c_s' h)) \\ \vdots & \ddots & \vdots \\ \widetilde{u_s'}(t + c_1' h) & \cdots & \widetilde{u_s'}(t + c_s' h) \end{pmatrix} \tag{3.17}$$

is non-singular, $A_{c_i,c_j'}$ are uniquely determined. Moreover if the nodes $0 < c_1, \dots, c_{s-1} < 1$ are different, the coefficients of $A_{\tau,\sigma}$ are also uniquely determined.

Obviously the non-singularity of the matrix (3.17) is equivalent to that of $M(t,h)$ in (3.6). Following the discussion there, we can conclude that if the Wronskian matrix is non-singular, the matrix (3.17) is also non-singular. This completes the proof. □

We consider order conditions. Let

$$T = \{\bullet, \bullet\!\!-\!\!\bullet, \bullet\!\!-\!\!\bullet\!\!-\!\!\bullet, \bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet, \bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet, \bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet, \bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet, \ldots\}$$

be the set of rooted trees. We denote the elementary weights by $\phi(\tau)$ ($\tau \in T$). Then the order conditions are summarised as follows.

Theorem 3.6. An EFCSRK method is of order p if

$$\phi(\tau) = \frac{1}{\gamma(\tau)} + \mathcal{O}(h^{p-|\tau|+1}) \quad \text{for} \quad \tau \in T, \quad |\tau| \leq p,$$

where $|\tau|$ denotes the order of τ.

3.4 Energy-preserving exponentially-fitted methods for Hamiltonian systems

In this section, we derive second and fourth order EPEFCSRK schemes for Hamiltonian systems. In what follows, we again consider $\mathcal{F}_1 = \{\exp(\lambda t), \exp(-\lambda t)\}$.

3.4.1 Second order EPEFCSRK scheme

Let us start with the one-degree CSRK formulation: $A_{\tau,\sigma} = a_{11}\tau$. In this case, the energy-preservation condition (3.13) and symmetry condition (3.14) are automatically satisfied. Therefore, the only thing we have to do is to determine the parameter a_{11} satisfying the EF conditions. When we consider the set $\mathcal{F}_1$, the linear system (3.16) reduces to

$$\mathrm{e}^{z} = 1 + a_{11} z \frac{1+\mathrm{e}^{z}}{2}, \quad \mathrm{e}^{-z} = 1 - a_{11} z \frac{1+\mathrm{e}^{-z}}{2},$$

where $z = \lambda h$. We can easily obtain

$$a_{11} = \frac{2\sinh(\frac{z}{2})}{z\cosh(\frac{z}{2})},$$

and the resulting scheme reads

$$y_1 = y_0 + a_{11} h \int_0^1 f((1-\sigma)y_0 + \sigma y_1)\,\mathrm{d}\sigma.$$

When we implement the scheme, if the value $|z|$ is small, the following series expansion should be used:

$$a_{11} = 1 - \frac{1}{12}z^2 + \frac{1}{120}z^4 - \frac{17}{20160}z^6 + \frac{31}{362880}z^8 - \frac{691}{79833600}z^{10} + \frac{5461}{6227020800}z^{12} + \cdots.$$

It is clear that in the limit $z \to 0$ the standard AVF method is recovered.

3.4.2 Fourth order EPEFCSRK scheme

Let us start with the two-degree CSRK formulation:

$$A_{\tau,\sigma} = a_{11}\tau + a_{12}\tau\sigma + a_{21}\tau^2 + a_{22}\tau^2\sigma.$$

Firstly, we consider the energy-preservation and symmetry conditions. The energy-preservation condition (3.13) is equivalent to

$$a_{12} = 2a_{21},$$

and the symmetry condition (3.14) is equivalent to

$$a_{22}+2a_{21}=0,\quad a_{22}+a_{12}=0.$$

Therefore, it follows that a two-degree CSRK method whose coefficients are related by

$$\begin{array}{cc|cc} a_{11} & a_{12} & a_{11} & 2a_{21} \\ a_{21} & a_{22} & a_{21} & -2a_{21} \\ \hline b_1 & b_2 & a_{11}+a_{21} & 0 \end{array} \tag{3.18}$$

is energy-preserving and symmetric.

Next we consider the EF conditions. We write Y_τ as a linear combination of y_0, Y_c and y_1, i.e.,

$$Y_\tau = y_0\frac{(\tau-c)(\tau-1)}{c}+Y_c\frac{\tau(\tau-1)}{c(c-1)}+y_1\frac{\tau(\tau-c)}{1-c}.$$

Then the method becomes

$$\begin{aligned} Y_c &= y_0+h\int_0^1 A_{c,\sigma}f(Y_\sigma)\,\mathrm{d}\sigma,\\ y_1 &= y_0+h\int_0^1 B_\tau f(Y_\tau)\,\mathrm{d}\tau. \end{aligned}$$

Note that although y_1 is independent of c in standard CSRK methods, the parameter c plays an important role in EFCSRK methods.

When we consider the set $\mathcal{F}_1$, the linear systems (3.15) and (3.16) reduce to

$$\int_0^1 A_{c,\sigma}\widetilde{\cosh}(\sigma z)\,\mathrm{d}\sigma=\frac{\sinh(cz)}{z},\quad \int_0^1 A_{c,\sigma}\widetilde{\sinh}(\sigma z)\,\mathrm{d}\sigma=\frac{\cosh(cz)-1}{z}, \tag{3.19}$$

$$\int_0^1 B_\sigma\widetilde{\cosh}(\sigma z)\,\mathrm{d}\sigma=\frac{\sinh(z)}{z},\quad \int_0^1 B_\sigma\widetilde{\sinh}(\sigma z)\,\mathrm{d}\sigma=\frac{\cosh(z)-1}{z}, \tag{3.20}$$

where $z=\lambda h$, $\widetilde{\cosh}(\sigma z)$ denotes a polynomial of degree two which is a linear combination of $\cosh(0)$, $\cosh(cz)$ and $\cosh(z)$, and the similar notation is used for sinh. Since B_σ is independent of σ (see (3.18)), the second conditions (3.20) are equivalent to

$$\begin{aligned} B_\sigma\frac{(3c^2-4c+1)-\cosh(cz)+(3c^2-2c)\cosh(z)}{6c^2-6c} &= \frac{\sinh(z)}{z},\\ B_\sigma\frac{-\sinh(cz)+(3c^2-2c)\sinh(z)}{6c^2-6c} &= \frac{\cosh(z)-1}{z}. \end{aligned}$$

We obtain $c=1/2$ from the compatibility condition

$$\begin{aligned} &\frac{(3c^2-4c+1)-\cosh(cz)+(3c^2-2c)\cosh(z)}{\sinh(z)}=\frac{-\sinh(cz)+(3c^2-2c)\sinh(z)}{\cosh(z)-1}\\ \Leftrightarrow\quad &(2c-1)(1-\cosh(z))+2\sinh\Big(\frac{z}{2}\Big)\sinh\Big(\frac{(2c-1)z}{2}\Big)=0, \end{aligned}$$

and thus have

$$B_\sigma=a_{11}+a_{21}=\frac{6(\cosh(z)-1)}{z\big(4\sinh(\frac{z}{2})+\sinh(z)\big)}. \tag{3.21}$$

Using the relation (3.18) and $c = 1/2$, we can write $A_{c,\sigma}$ as

$$A_{1/2,\sigma} = \frac{a_{11}}{2} + \frac{a_{12}}{2}\sigma + \frac{a_{21}}{4} + \frac{a_{22}}{4}\sigma = \frac{a_{21}}{2}\sigma + \Big(\frac{a_{11}}{2} + \frac{a_{21}}{4}\Big).$$

From the first EF conditions (3.19), a_{11} and a_{21} are uniquely determined as follows.

$$a_{11} = \frac{6\big(-7 + 4\cosh(\frac{z}{2}) + 3\cosh(z)\big)}{z\big(4\sinh(\frac{z}{2}) + \sinh(z)\big)}, \tag{3.22}$$

$$a_{21} = \frac{12\big(3 - 2\cosh(\frac{z}{2}) - \cosh(z)\big)}{z\big(4\sinh(\frac{z}{2}) + \sinh(z)\big)}. \tag{3.23}$$

Obviously they satisfy (3.21). Therefore, we can conclude that the EF conditions (3.19) and (3.20) are compatible if and only if $c = 1/2$, and the relations (3.22) and (3.23) are satisfied. Note that for the numerical computation, the following series expansions should be employed:

$$\begin{aligned} a_{11} &= 4 - \frac{1}{16}z^2 + \frac{7}{5760}z^4 - \frac{113}{3870720}z^6 + \frac{79}{92897280}z^8 - \frac{229}{11678515200}z^{10} \\ &\quad + \frac{55067}{76517631590400}z^{12} + \cdots, \\ a_{21} &= -3 + \frac{1}{16}z^2 - \frac{1}{640}z^4 + \frac{17}{430080}z^6 - \frac{31}{30965760}z^8 + \frac{691}{27249868800}z^{10} \\ &\quad - \frac{5461}{8501959065600}z^{12} + \cdots. \end{aligned}$$

It is clear that in the limit $z \to 0$ the standard fourth order EP method (2.15) is reproduced.

The derived scheme has at least the accuracy of order two because it is symmetric. Moreover since the coefficients satisfy the conditions for order three (Theorem 3.6)

$$\begin{aligned} \int_0^1\int_0^1\int_0^1 B_\sigma A_{\sigma,\tau}A_{\sigma,\nu}\,\mathrm{d}\sigma\mathrm{d}\tau\mathrm{d}\nu &= \frac{1}{3} + \mathcal{O}(z^2), \\ \int_0^1\int_0^1\int_0^1 B_\sigma A_{\sigma,\tau}A_{\tau,\nu}\,\mathrm{d}\sigma\mathrm{d}\tau\mathrm{d}\nu &= \frac{1}{6} + \mathcal{O}(z^2), \end{aligned}$$

the scheme has the accuracy of order four.

3.4.3 Numerical examples

We present two numerical experiments that confirm the effectiveness of the EPEFCSRK methods. Through some typical Hamiltonian problems, we compare the EPEFCSRK methods with the standard EPCSRK methods. Note that although in the previous subsections we considered $\mathcal{F}_1 = \{\exp(\lambda t), \exp(-\lambda t)\}$, we can automatically obtain EF schemes for $\mathcal{F}_2 = \{\cos(\omega t), \sin(\omega t)\}$ by substituting $\lambda = i\omega$. In the following examples, we consider problems where we know a suitable value of the parameter ω in advance. Of course, the choice of the parameter ω is one of the important issues in the use of EF methods. Several standard ways are summarised in [183, Section 6]. For the numerical experiments, we used MATLAB. Nonlinear equations were solved by the simplified Newton iteration with tolerance 10^{-12}.

The Kepler problem

We consider the Kepler two-body problem defined by the Hamiltonian

$$H(q,p) = \frac{p_1^2 + p_2^2}{2} - \frac{1}{\sqrt{q_1^2 + q_2^2}},$$

with the initial conditions $q_1 = 1-e$, $q_2 = 0$, $p_1 = 0$, $p_2 = \sqrt{(1+e)/(1-e)}$, where e $(0 \le e \le 1)$ represents the eccentricity of the elliptic orbit. In the numerical experiments, we have set the values to $e = 0.02$, $\lambda = i\omega$ with $\omega = (q_1^2 + q_2^2)^{-3/2}$.

Before showing the numerical results, we mention some implementation issues. For energy-preserving methods, we have to exactly compute the average of the vector field, i.e., the right hand side of (3.11) and (3.12) before the implementation. Although we can integrate them exactly for second order schemes, we cannot do that for fourth order ones (in the sense that mathematica could not calculate them exactly). Instead, we have integrated them numerically, using *integral* function in MATLAB with the tolerance 10^{-12}.

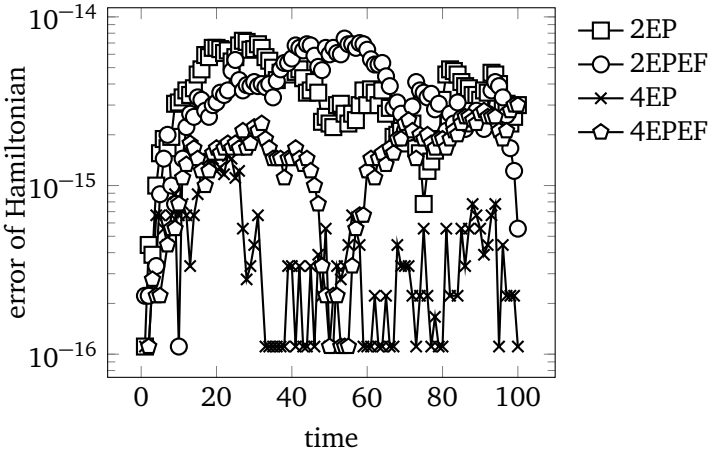


Figure 3.1: The variation of the errors of Hamiltonian for the Kepler problem ($e = 0.02$). The stepsize was set to $h = 0.05$.

The variations of the error of the Hamiltonian are shown in Figure 3.1. All schemes preserve the Hamiltonian well.

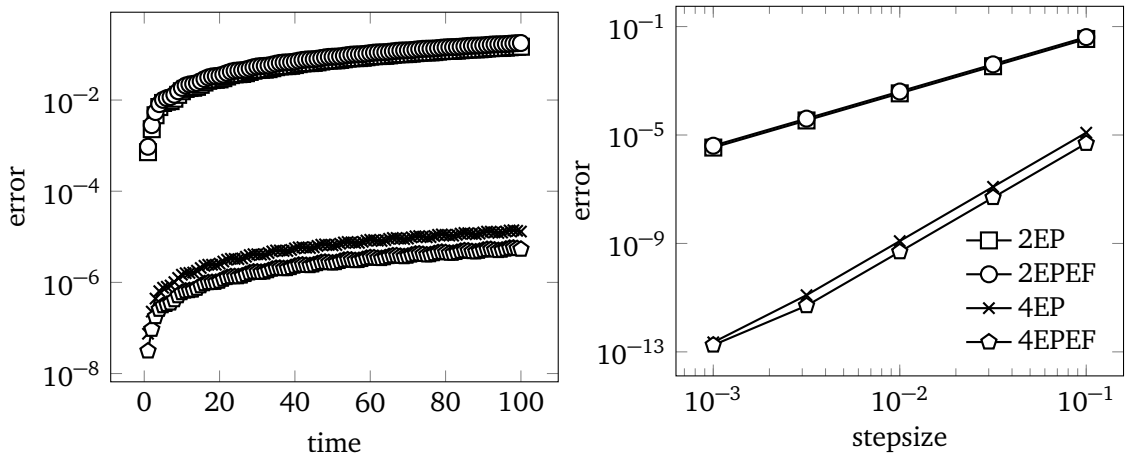


Figure 3.2: LEFT: The variation of the maximum global errors for the Kepler problem ($e = 0.02$). The stepsize was set to $h = 0.05$. RIGHT: The global errors at $t = 5$.

In Figure 3.2 (left) one can see that the errors are growing linearly with time for all four schemes. The result by the second order EPEFCSRK scheme is more or less the same as that by the standard second order EPCSRK scheme. But the result by the fourth order EPEFCSRK scheme is better than that by the standard fourth order EPCSRK scheme. Figure 3.2 (right) shows the convergence of the

numerical solutions. One can observe the expected convergence rates in all the schemes.

In this example, we used a very small eccentricity $e = 0.02$ so that the solution is close to the circle. Numerical behaviour with the large eccentricity is also of interest. We observed that the fourth order EPEFCSRK scheme gives better numerical solutions than others if e is up to around 0.45. .

An oscillatory Hamiltonian problem with cubic nonlinearity

We consider an oscillatory Hamiltonian problem with cubic nonlinearity defined by

$$H(q,p) = \frac{1}{2}p^2 + \frac{1}{2}\omega^2 q^2 - \frac{1}{4}q^4.$$

with $\omega = 10$ and the initial condition $(q,p) = (1.5, 0)$.

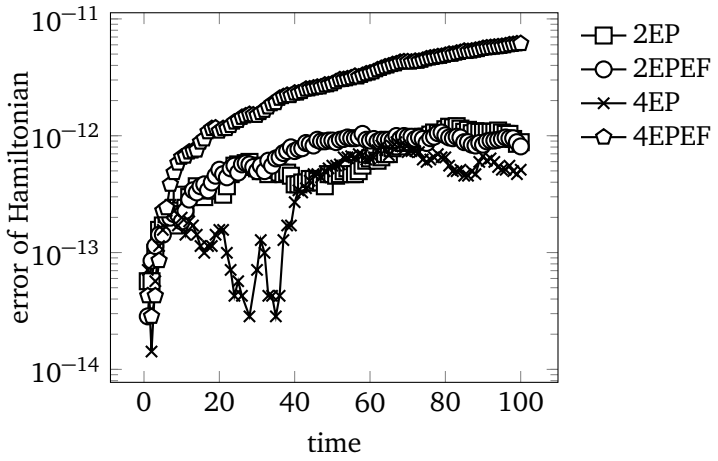


Figure 3.3: The variation of the errors of Hamiltonian for the oscillatory Hamiltonian problem. The stepsize was set to $h = 0.05$.

The variations of the error of the Hamiltonian are shown Figure 3.3. All schemes preserve the Hamiltonian well. The error of fourth order EPEFCSRK scheme seems to grow linearly due to rounding errors. Such errors might be controlled by the technique presented in [98] or by setting a new stopping criteria for the (simplified) Newton method in terms of the Hamiltonian.

In Figure 3.4 (left) one can see that the errors are growing linearly with time for all four schemes. The results by the EPEFCSRK schemes are better than those by the standard EPCSRK schemes. Figure 3.4 (right) shows the convergence of the numerical solutions. One can observe the expected convergence rates in all of the schemes.

Discussions

The advantage of using EF methods is much more remarkable for the oscillatory Hamiltonian problem than for the Kepler problem. The main reason would be that the period of the oscillatory Hamiltonian problem is smaller then that of the Kepler problem, and moreover the oscillatory Hamiltonian problem behaves like the simple harmonic oscillator.

In order to obtain a more good performance for the Kepler problem, it is possible to use different ω values at different time steps. However, this approach might deteriorate the quality of the numerical solution over a long-time interval, because the excellent long-time behaviour is usually guaranteed based on the assumption that we do not change the integrator or stepsize during the computation. An alternative way would be to construct energy-preserving integrators which exactly solve elliptic solutions. However, it is still open if this approach is possible.

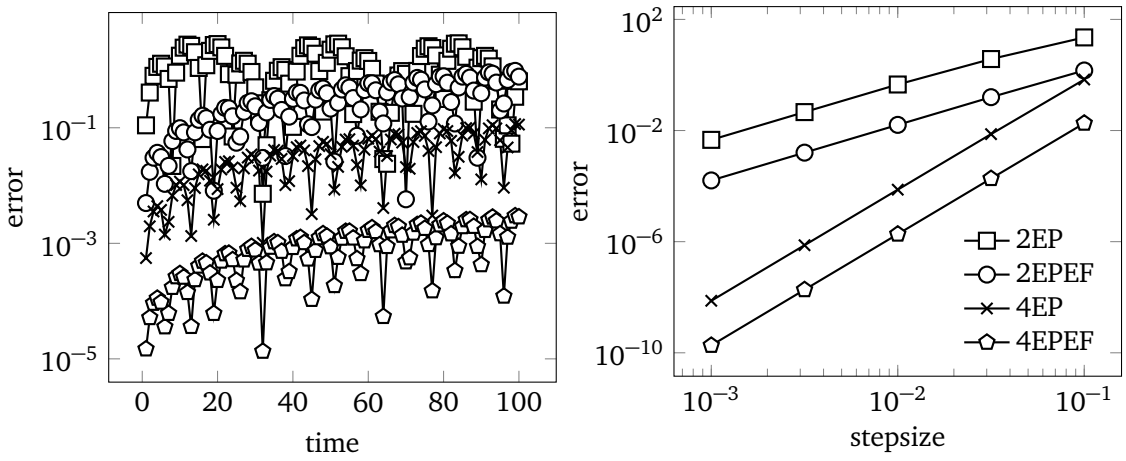


Figure 3.4: LEFT: The variation of the maximum global errors for the oscillatory Hamiltonian problem. The stepsize was set to $h = 0.05$. RIGHT: The global errors at $t = 5$.

3.5 Energy-preserving exponentially-fitted methods for Poisson systems

We shall derive second- and fourth-order energy-preserving EF schemes for Poisson systems. While the derivation of second-order scheme is relatively easy, the derivation of fourth-order one requires a more careful treatment.

3.5.1 Second order scheme

Let $s = 1$. In this case, we have $c = 1/2$ from the symmetry condition. The PCSRK method reduces to

$$y_1 = y_0 + a_{11} h \Lambda \Big(\frac{y_0 + y_1}{2} \Big) \int_0^1 \nabla H(y_0 + \tau(y_1 - y_0)) \, \mathrm{d}\tau$$

with a parameter a_{11}. This method is always energy-preserving independently of a_{11}. When one considers the set $\mathcal{F}_1 = \{\exp(\lambda t), \exp(-\lambda t)\}$, the EF condition is given as

$$e^z = 1 + a_{11} z \frac{1 + e^z}{2}, \quad e^{-z} = 1 - a_{11} z \frac{1 + e^{-z}}{2},$$

where $z = \lambda h$, from which we immediately obtain

$$a_{11} = \frac{2 \sinh(\frac{z}{2})}{z \cosh(\frac{z}{2})}.$$

The resulting scheme reads

$$y_1 = y_0 + \frac{2 \sinh(\frac{z}{2})}{z \cosh(\frac{z}{2})} h \Lambda \Big(\frac{y_0 + y_1}{2} \Big) \int_0^1 \nabla H(y_0 + \tau(y_1 - y_0)) \, \mathrm{d}\tau.$$

3.5.2 Fourth order scheme

A more interesting, nontrivial example is the derivation of fourth order schemes. Setting $s = 2$, we consider coefficient polynomials of the form $A_{i\tau, j\sigma} = a^j_{11}\tau + a^j_{12}\tau\sigma + a^j_{21}\tau^2 + a^j_{22}\tau^2\sigma$ $(j = 1, 2)$. Our aim is to determine these eight parameters and two nodes c_1, c_2 (thus, there are 10 unknowns) so that they satisfy conditions of energy-preservation, symmetry, exponential-fitting, and order. Note that considering symmetry conditions makes the derivation simple, that is, we do not have to care conditions for odd orders. The procedure consists of the following four steps.

Step 1 (Energy-preservation condition) From Theorem 3.4, the energy-preservation condition is equivalent to $a^j_{12} = 2a^j_{21}$ $(j = 1, 2)$.

Step 2 (Symmetry condition) From Theorem 3.5, the symmetry condition is equivalent to

$$a^1_{22} = a^2_{22}, \quad a^1_{21} + a^2_{21} = -a^1_{22}, \quad a^1_{11} - a^2_{11} = 4a^2_{21} + 2a^2_{22}, \quad c_1 + c_2 = 1.$$

Step 3 (EF condition) Next we consider the EF conditions, which make the scheme exact for ODEs whose solution belongs to $\mathcal{F}_1 = \{\exp(\lambda t), \exp(-\lambda t)\}$. Note that there is a linear ODE whose solution belongs to this space. Thus, we can reduce the discussion of the EF conditions into the framework of CSRK methods with $A_{\tau,\sigma} = \sum_{j=1}^{2} A_{i\tau,j\sigma}$. From the discussion in Section 3.4.2, the parameters of $A_{\tau,\sigma}$ are given as (3.22) and (3.23). Hence, for the coefficients of PCSRK methods, we have

$$a^1_{11} + a^2_{11} = \frac{6(-7 + 4\cosh(\frac{z}{2}) + 3\cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))}, \quad a^1_{21} + a^2_{21} = \frac{12(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))}.$$

Step 4 (Order condition) Finally, we consider the order conditions. Let

$$TP = \{\bullet, \circ, \dots\}$$

be a set of rooted bi-coloured trees. We denote the subset of TP whose roots are $\bullet$ by TP_y, and the remains by TP_z. We also denote the elementary weights by $\phi(\tau)$ $(\tau \in TP)$. Since the elementary weights depend on h for EF methods, we have to consider the order conditions taking this effect into consideration. The order conditions are summarised as follows. An EF PCSRK method is of order p if

$$\phi(\tau) = \frac{1}{\gamma(\tau)} + \mathcal{O}(h^{p-|\tau|+1}) \quad \text{for} \quad \tau \in TP_y, \quad |\tau| \le p,$$

where $|\tau|$ denotes the order of τ (i.e., the number of vertices). Note that we do not have to consider TP_z because $y_1 = z_1$. Since symmetric methods always have even order, it is sufficient to consider only first- and third-order conditions:

$$\begin{aligned}
&\phi(\bullet) = 1 + \mathcal{O}(h^4),\\
&\phi(\dots) = \frac{1}{3} + \mathcal{O}(h^2), \quad \phi(\dots) = \frac{1}{3} + \mathcal{O}(h^2), \quad \phi(\dots) = \frac{1}{3} + \mathcal{O}(h^2),\\
&\phi(\dots) = \frac{1}{6} + \mathcal{O}(h^2), \quad \phi(\dots) = \frac{1}{6} + \mathcal{O}(h^2), \quad \phi(\dots) = \frac{1}{6} + \mathcal{O}(h^2), \quad \phi(\dots) = \frac{1}{6} + \mathcal{O}(h^2).
\end{aligned}$$

These conditions are all satisfied if the perturbations of all parameters from those of the standard fourth-order energy-preserving scheme

$$A_{i\tau,1\sigma} = \frac{1}{2\sqrt{3}}\left((4\sqrt{3}+6)\tau - 6(1+\sqrt{3})\tau\sigma - 3(1+\sqrt{3})\tau^2 + 6\sqrt{3}\tau^2\sigma\right), \tag{3.24}$$

$$A_{i\tau,2\sigma} = \frac{1}{2\sqrt{3}}\left((4\sqrt{3}-6)\tau + 6(1-\sqrt{3})\tau\sigma + 3(1-\sqrt{3})\tau^2 + 6\sqrt{3}\tau^2\sigma\right), \tag{3.25}$$

$$c_{1,2} = \frac{1}{2} \mp \frac{\sqrt{3}}{6} \tag{3.26}$$

are less than $\mathcal{O}(h^2)$. Note that we have obtained the eight independent conditions for the ten parameters by Step 3, and thus two freedoms still remain at this stage. If one introduces two additional constraints arbitrarily so that they are consistent with the above order conditions, all parameters are uniquely determined.

In the next section, we simply consider the choices

$$c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}, \qquad 2a_{21} + a_{22} = -\sqrt{3}$$

as additional constraints in Step 4. Then all parameters are uniquely determined to be

$$\begin{aligned}
a^1_{11} &= \frac{3(-7 + 4\cosh(\frac{z}{2}) + 3\cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} + \sqrt{3},\\
a^1_{12} &= \frac{12(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} - \sqrt{3},\\
a^1_{21} &= \frac{6(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} - \frac{\sqrt{3}}{2},\\
a^1_{22} &= -\frac{12(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))},\\
a^2_{11} &= \frac{3(-7 + 4\cosh(\frac{z}{2}) + 3\cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} - \sqrt{3}\\
a^2_{12} &= \frac{12(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} + \sqrt{3},\\
a^2_{21} &= \frac{6(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))} + \frac{\sqrt{3}}{2},\\
a^2_{22} &= -\frac{12(3 - 2\cosh(\frac{z}{2}) - \cosh(z))}{z(4\sinh(\frac{z}{2}) + \sinh(z))},\\
c_1 &= \frac{1}{2} - \frac{\sqrt{3}}{6},\\
c_2 &= \frac{1}{2} + \frac{\sqrt{3}}{6}.
\end{aligned}$$

It is checked that the perturbations of these parameters from those in (3.24), (3.25) and (3.26) are less than $\mathcal{O}(h^2)$.

3.5.3 Numerical examples

We test the derived schemes numerically. For a problem whose period is estimated to $T = 2\pi/\omega$, we consider the set $\mathcal{F}_1 = \{\exp(\lambda t), \exp(-\lambda t)\}$ with $\lambda = i\omega$, which is equivalent to $\{\sin(\omega t), \cos(\omega t)\}$. We used Python and its numpy and scipy packages.

We consider the Euler equation

$$\dot{q} = f(q) = ((\alpha - \beta)q_2 q_3, (1 - \alpha)q_3 q_1, (\beta - 1)q_1 q_2)^\top,$$

which describes the motion of a rigid body under no forces. This system can be seen as the Poisson system

$$\dot{q} = \begin{pmatrix} 0 & \alpha q_3 & -\beta q_2 \\ -\alpha q_3 & 0 & q_1 \\ \beta q_2 & -q_1 & 0 \end{pmatrix} \nabla H(q), \quad H(q) = \frac{q_1^2 + q_2^2 + q_3^2}{2}.$$

We set the initial value to $q(0) = (0, 1, 1)^\top$, and the parameters to $\alpha = 1 + (1/\sqrt{1.51})$, $\beta = 1 - (0.51/\sqrt{1.51})$, which are employed in [38]. The exact solution is given by

$$q(t) = (\sqrt{1.51}\,\mathrm{sn}(t, 0.51), \mathrm{cn}(t, 0.51), \mathrm{dn}(t, 0.51))^\top,$$

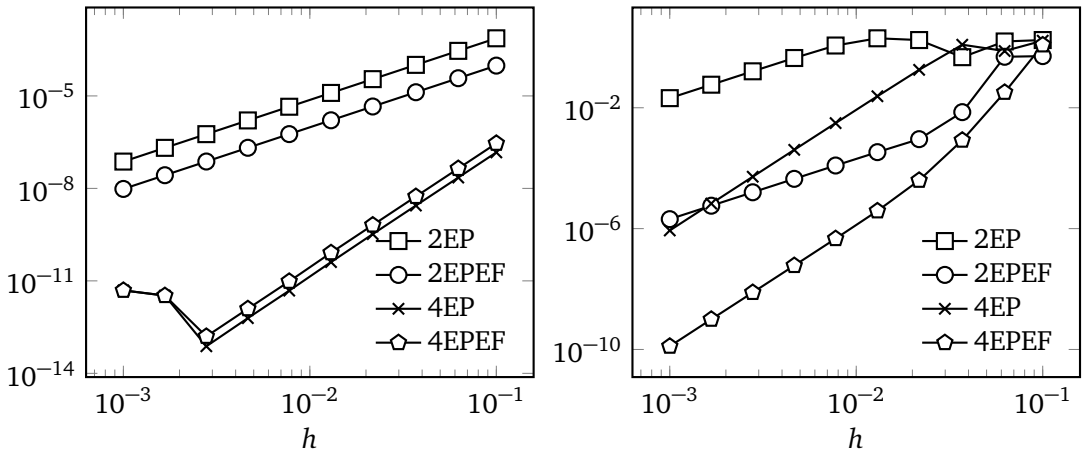


Figure 3.5: Global errors at $t = 2$ of (left) the first and (right) second examples for Euler equations. 2EP: standard second-order energy-preserving scheme, 2EPEF: second-order energy-preserving EF scheme, 4EP: standard fourth-order energy-preserving scheme, 4EPEF: fourth-order energy-preserving EF scheme.

where $\mathrm{sn}, \mathrm{cn}, \mathrm{dn}$ are the Jacobi elliptic functions [38]. This solution is periodic with the period $T = 4K(0.51) = 7.450563209330954$, where $K(k)$ stands for the complete elliptic integral of the first kind defined by

$$K(k) = \int_0^{\pi/2} \frac{1}{\sqrt{1-k^2\sin^2\theta}}\,\mathrm{d}\theta = \int_0^1 \frac{1}{\sqrt{(1-t^2)(1-k^2t^2)}}\,\mathrm{d}t.$$

Figure 3.5 (left) plots the global error, from which one can see that the solution by the second-order energy-preserving EF scheme is better than that by the standard second-order energy-preserving scheme. Despite the expectation, the results by the fourth-order EF scheme is worse than the standard fourth-order scheme. The reason would be that the period of the solution is relatively large. As already mentioned in the previous section, it is still open if the energy-preserving methods can be fitted to elliptic functions.

We also consider a more anomalous case. When $\beta \approx 1$, we expect, at least intuitively, $\dot{q}_3 \approx 0$ and thus $q_3(t) \approx 1$. Therefore, the variables q_1 and q_2 seem to behave like harmonic oscillator with period $T = 2\pi/(\alpha-1)$. We set $\alpha = 51$ and $\beta = 1.01$. The global error is shown in Figure 3.5 (right). One can see that EF schemes produce much better solutions than the same order, standard energy-preserving schemes.

For the second problem, the error of the Hamiltonian obtained by the fourth-order energy-preserving EF scheme is plotted in Figure 3.6. In addition, the error of another invariant $I = (q_1^2 + \beta q_2^2 + \alpha q_3^2)/2$ is plotted. It is observed that the second invariant is nearly preserved without any drift.

3.6 Explicit methods

We saw in Section 2.5.4 that an explicit energy-preserving integrator can be constructed for Hamiltonian systems with Hamiltonian

$$H(q,p) = \frac{1}{2}p^\top p + \frac{1}{2}q^\top A q + U(q),$$

where A is a positive semi-definite matrix. A corresponding differential equation reads

$$\dot{q} = p, \qquad \dot{p} = -Aq - \nabla U(q),$$

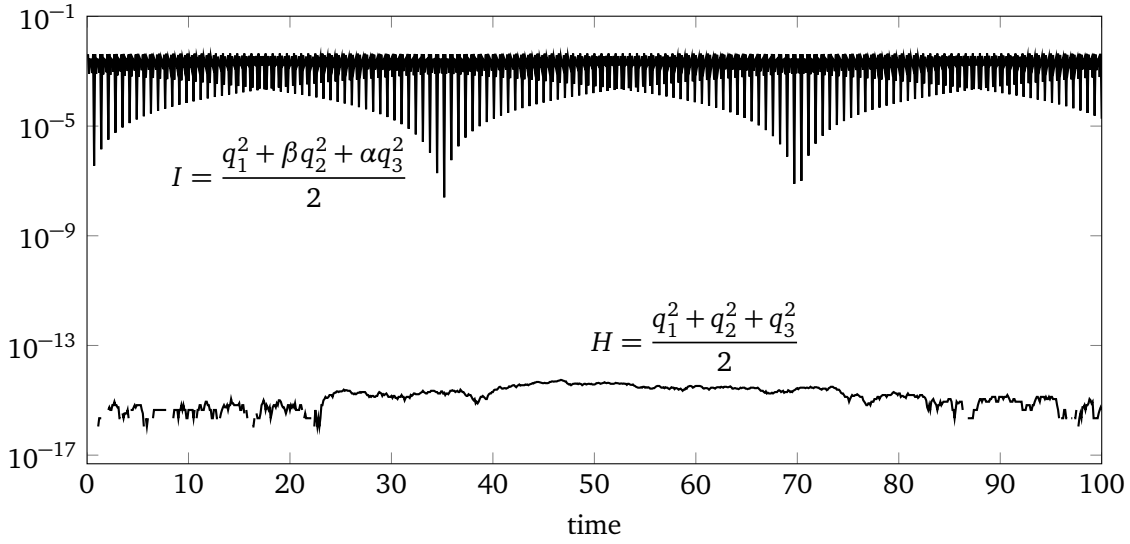


Figure 3.6: Errors of two conserved quantities obtained by the fourth-order energy-preserving EF scheme with the stepsize $h = 0.1$.

or

$$\ddot{q} + Aq = g(q), \qquad \text{where} \qquad g(q) = -\nabla U(q).$$

Recall that the explicit energy-preserving integrator does not preserve the energy exactly, but preserve the modified version exactly. Here, we consider the situation where A is a positive semi-definite constant matrix (so that $A = \Omega^2$) with large norm. In this case, the system exhibits oscillatory behaviour. In this section, by modifying the scheme (2.17), we shall derive explicit energy-preserving integrators, which capture the oscillation by a large stepsize. To this aim, we use the idea of the so called trigonometric method.

3.6.1 A brief review of trigonometric methods

Trigonometric methods are briefly reviewed. For more details, see [97, Chapter XIII].

The Störmer–Verlet method (2.12) often gives qualitatively nice numerical solutions with a relatively large stepsize h, but we cannot use a large stepsize for oscillatory Hamiltonian systems because of the explicitness of the method. For a while, just for simplicity, let us consider a one-dimensional problem with $\Omega = \omega \in \mathbb{R}$. The linear stability analysis tells us that the method is stable only if $h = \mathcal{O}(\omega^{-1})$. In particular, when $g = 0$, we have to chose a stepsize so that $h\omega \leq 2$. This drawback is overcome by trigonometric methods.

The main idea of trigonometric methods is modifying the Störmer–Verlet method so that they are exact for linear problems with $g(q) = -\nabla U(q) = 0$. This aim is made possible by the two-step form

$$q_{n+1} - 2\cos(h\Omega)q_n + q_{n-1} = 0. \tag{3.27}$$

When $g(q) \neq 0$, the most natural extension seems $q_{n+1} - 2\cos(h\Omega)q_n + q_{n-1} = h^2 g(q_n)$, but several modifications have been considered. They can be written in the form

$$q_{n+1} - 2\cos(h\Omega)q_n + q_{n-1} = h^2 \Psi g(\Phi q_n)$$

with $\Psi = \psi(h\Omega)$ and $\Phi = \phi(h\Omega)$, where the filter functions ψ and ϕ are even and real-valued with $\psi(0) = \phi(0) = 1$. Here are several choices of the filter functions ($\operatorname{sinc}\xi := \frac{\sin\xi}{\xi}$). The alphabetical order follows [97, Chapter XIII.2.2], and the method (F) is omitted because it is a two-force method.

(A)	$\psi(\xi)=\operatorname{sinc}^2(\frac{1}{2}\xi)$	$\phi(\xi)=1$	Gautschi [86] (1961)
(B)	$\psi(\xi)=\operatorname{sinc}(\frac{1}{2}\xi)$	$\phi(\xi)=1$	Deuflhard [68] (1979)
(C)	$\psi(\xi)=\operatorname{sinc}^2(\xi)$	$\phi(\xi)=\operatorname{sinc}(\xi)$	García-Archilla et al. [85] (1998)
(D)	$\psi(\xi)=\operatorname{sinc}^2(\frac{1}{2}\xi)$	$\phi(\xi)=\operatorname{sinc}\xi(1+\frac{1}{3}\sin^2(\frac{1}{2}\xi))$	Hochbruck–Lubich [104] (1999)
(E)	$\psi(\xi)=\operatorname{sinc}^2(\xi)$	$\phi(\xi)=1$	Hairer–Lubich [95] (2000)
(G)	$\psi(\xi)=\operatorname{sinc}^3(\xi)$	$\phi(\xi)=\operatorname{sinc}(\xi)$	Grimm–Hochbruck [91] (2006)

For the motivation of each choice, see each reference.

In addition to the initial value q_0, we have to compute q_1 in advance. As this was done for the Störmer–Verlet method in Section 2.4.1, the formula (3.27) can be written in the one-step form

$$q_{n+1}=\cos(h\Omega)q_n+\Omega^{-1}\sin(h\Omega)p_n+\frac{1}{2}h^2\Psi g_n, \tag{3.28}$$

$$p_{n+1}=-\Omega\sin(h\Omega)q_n+\cos(h\Omega)p_n+\frac{1}{2}h(\Psi_0 g_n+\Psi_1 g_{n+1}), \tag{3.29}$$

where $g_n=g(\Phi q_n)$ and $\Phi_0=\psi_0(h\Omega)$, $\Phi_1=\psi_1(h\Omega)$ with functions ψ_0, ψ_1 satisfying

$$\psi(\xi)=\operatorname{sinc}(\xi)\psi_1(\xi),\qquad \psi_0(\xi)=\cos(\xi)\psi_1(\xi).$$

The one-step form (3.28), (3.29) is interpreted as follows. The exact solution of the system can be expressed as the variation-of-constants formula

$$\begin{pmatrix}q(t)\\p(t)\end{pmatrix}=\begin{pmatrix}\cos t\Omega & \Omega^{-1}\sin t\Omega\\ -\Omega\sin t\Omega & \cos t\Omega\end{pmatrix}\begin{pmatrix}q_0\\p_0\end{pmatrix}+\int_0^t\begin{pmatrix}\Omega^{-1}\sin(t-s)\Omega\\ \cos(t-s)\Omega\end{pmatrix}g(q(s))\,\mathrm{d}s. \tag{3.30}$$

Thus, $\frac{1}{2}h^2\Psi g_n$ and $\frac{1}{2}h(\Psi_0 g_n+\Psi_1 g_{n+1})$ can be seen as results of discretising the integral in the variation-of-constants formula.

Grimm–Hochbruck [91] considered the error estimate of the one-step form (3.28), (3.29). Similar results are proved in [97, Chapter XIII] by a different technique.

3.6.2 Energy-preserving trigonometric integrators

As is the case with the Störmer–Verlet method, the explicit energy-preserving integrator (2.17) cannot solve highly oscillatory Hamiltonian systems efficiently. Then, based on the scheme (2.17), we consider a class of trigonometric methods of the form

$$q_{n+1}+(1-2\cos(h\Omega))(q_n+q_{n-1})+q_{n-2}=-2h^2\Psi\overline{\nabla}U(\Phi q_n,\Phi q_{n-1}), \tag{3.31}$$

so that it exactly solves Hamiltonian systems when $\nabla U=0$. Here, $\overline{\nabla}U$ denotes the discrete gradient of U.

Theorem 3.7. The integrator (3.31) is energy-preserving in the sense that

$$\frac{1}{2}\Big(\frac{q_{n+1}-q_n}{h}\Big)^{\top}\Big(\frac{q_n-q_{n-1}}{h}\Big)+\operatorname{sinc}^2(h\Omega/2)\Big(\frac{1}{2}q_n^{\top}Aq_n+\Psi U(\Phi q_n)\Big)=\text{const.}$$

Proof. Note that the integrator (3.31) is derived based on the Lagrangian

$$L_{\mathrm{d}}(q_n,\delta_t^+q_n):=\frac{1}{2\operatorname{sinc}^2(h\Omega/2)}(\delta_t^+q_n)^{\top}(\delta_t^+q_n)-\frac{1}{2}q_n^{\top}\Omega^2q_n-U(q_n).$$

Then, by construction, the energy-preservation property immediately follows. □

Because the integrator (3.31) is of a three-step form, we have to compute q_1 and q_2 in advance. Below, to this end, we rewrite (3.31) in a one-step form. First, we show the idea of this reformulation for the basic integrator (2.17). We introduce approximations of $p = \dot{q}$ and $r = \ddot{q}$:

$$p_n = \frac{q_{n+1} - q_{n-1}}{2h}, \qquad r_n = \frac{q_{n+1} - 2q_n + q_{n-1}}{h^2}.$$

We supplementarily use

$$p_{n+1/2} = \frac{q_{n+1} - q_n}{h}.$$

Then, writing (2.17) as

$$p_{n+1} - p_n = -h\overline{\nabla}U(q_{n+1}, q_n)$$

or

$$r_{n+1} + r_n = -2\overline{\nabla}U(q_{n+1}, q_n)$$

and using the relations

$$p_{n+1/2} - p_{n-1/2} = hr_n, \qquad p_{n+1/2} + p_{n-1/2} = 2p_n,$$

we get the formula

$$\begin{aligned} p_{n+1/2} &= p_n + \frac{h}{2} r_n, \\ q_{n+1} &= q_n + h p_{n+1/2}, \\ p_{n+1} &= p_n - h\overline{\nabla}U(q_{n+1}, q_n), \\ r_{n+1} &= -r_n - 2\overline{\nabla}U(q_{n+1}, q_n). \end{aligned}$$

Eliminating $p_{n+1/2}$ leads to

$$\begin{aligned} q_{n+1} &= q_n + h\left(p_n + \frac{h}{2} r_n\right), \\ p_{n+1} &= p_n - h\overline{\nabla}U(q_{n+1}, q_n), \\ r_{n+1} &= -r_n - 2\overline{\nabla}U(q_{n+1}, q_n). \end{aligned}$$

Next, let us consider a similar discussion for the trigonometric integrator (3.31). We introduce approximations of $p = \dot{q}$ and $r = \ddot{q}$:

$$p_n = \frac{q_{n+1} - q_n}{2h\,\mathrm{sinc}(h\Omega)}, \qquad r_n = \frac{q_{n+1} - 2q_n + q_{n-1}}{h^2\,\mathrm{sinc}^2(h\Omega/2)}.$$

By supplementarily using

$$p_{n+1/2} = \frac{q_{n+1} - q_n}{h\,\mathrm{sinc}(h\Omega/2)},$$

we finally obtain

$$q_{n+1} = q_n + \Omega^{-1}\sin(h\Omega)p_n + \frac{h^2}{2}\mathrm{sinc}^2(h\Omega/2)r_n, \tag{3.32}$$

$$p_{n+1} = -4\Omega\frac{\sin^2(h\Omega/2)}{\sin(h\Omega)}q_n + \cos(h\Omega)p_n - h\frac{\sin^2(h\Omega/2)\,\mathrm{sinc}^2(h\Omega/2)}{\mathrm{sinc}(h\Omega)}r_n - h\frac{\Psi}{\mathrm{sinc}(h\Omega)}\overline{\nabla}U(\Phi q_{n+1}, \Phi q_n), \tag{3.33}$$

$$r_{n+1} = -2\Omega^2 q_n - \Omega\sin(h\Omega)p_n - (1 + 2\sin^2(h\Omega/2))r_n - \frac{2\Psi}{\mathrm{sinc}(h\Omega)}\overline{\nabla}U(\Phi q_{n+1}, \Phi q_n). \tag{3.34}$$

The initial value of r_n is computed by either $r_0 = -\Omega^2 q_0 - \nabla U(q_0)$ or $r_0 = -\Omega^2 q_0 - \Psi \nabla U(\Phi q_0)$.

Note that the first term of the right hand side of (3.33) is not bounded for all h. In fact, the term diverges when $h\omega = 2k\pi$ ($k \in \mathbb{N}$). In such a case, we adopt the three-step form (3.31) by preparing $q_{\pm 1}$ based on (3.32). In a similar way, p_n can also be computed by a three-step form

$$p_{n+1} + (1 - 2\cos(h\Omega))(p_n + p_{n-1}) + p_{n-2} = -\frac{h\Psi}{\operatorname{sinc}(h\Omega)}\big(\overline{\nabla} U(\Phi q_{n+1}, \Phi q_n) - \overline{\nabla} U(\Phi q_{n-1}, \Phi q_{n-2})\big).$$

The starting values $p_{\pm 1}$ is obtained by

$$\begin{aligned} p_{\pm 1} = {} & \mp \Omega \sin(h\Omega) q_0 + \cos(h\Omega) p_0 \\ & \pm h \sin^2(h\Omega/2) \operatorname{sinc}^2(h\Omega/2) \frac{\Psi}{\operatorname{sinc}(h\Omega)} \nabla U(\Phi q_0) \mp h \frac{\Psi}{\operatorname{sinc}(h\Omega)} \overline{\nabla} U(\Phi q_0, \Phi q_{\pm 1}). \end{aligned}$$

3.6.3 Numerical examples and discussions

We apply the proposed integrators to the Fermi–Pasta–Ulam problem (Example 2.2). Figure 3.7 shows the energy exchanges of the standard energy-preserving integrator (2.17) and the energy-preserving trigonometric integrators (3.31) with the choices of the filter functions (A)–(G) (see Figure 2.3 for the exact solution).

The results of the standard integrator is stable in the sense that the energies do not diverge, because $h\omega \leq 2$ is satisfied. However, the oscillation of the time scale $\mathcal{O}(\omega^{-1})$ is much bigger than that of the exact solution. On the other hand, the scales of the oscillations of the methods (3.31) with (A)–(G) are similar to the exact solution. In this sense, these methods with (A)–(G) are more practical than the standard energy-preserving method. However, these methods also behave differently from each other, and it is observed that only the methods (B) and (D) produce a nice approximation of the energy exchange.

In the above example, the methods (3.31) with (B) and (D) seem good, but they also have the drawback: clearly, the two methods do not give a good numerical solution when $h\omega$ is close to an integer multiples of π, because some fractions appearing in (3.32) and (3.33) become quite large. This phenomena is problematic especially for problems with several constant frequencies, because after all $h = \mathcal{O}(\omega^{-1})$ is necessary.

For the methods (3.31), the following should be further investigated. First, the convergence analysis is still missing. The reason is that while the analysis of the method (3.28) and (3.29) is based on the observation that it can be regarded as a discretisation of the variation-of-constants formula (3.30), the relation between the new methods (3.31) (and their one-step formulations (3.32)–(3.34)) and the variation-of-constants formula is not clear. Second, it would be interesting to search a choice of the filter functions, which not only gives a good approximation of the energy exchange but also remains in a good approximation even when $h\omega$ is close to an integer multiples of π. Finally, it is still not clear under what circumstances (or for what kind of problems) the new methods (3.31) is superior to the standard trigonometric methods (3.27). They should also be further studied.

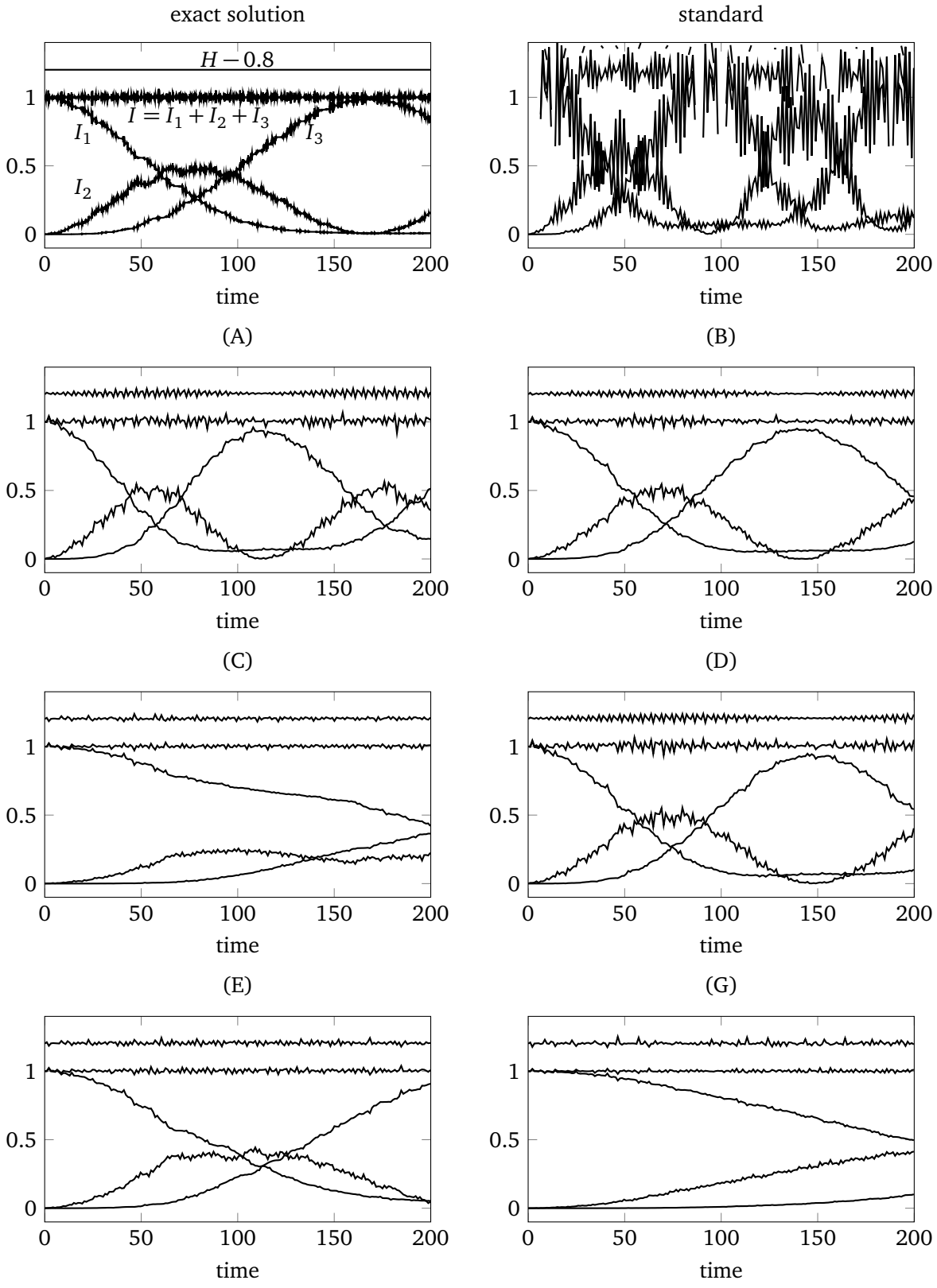


Figure 3.7: The Hamiltonian and oscillatory energies ($H-0.8$, I, I_1, I_2, I_3) obtained by several integrators, as well as the exact solution, for the Fermi–Past–Ulam problem are plotted. The parameters are set to $m = 3$ and $\omega = 50$. The initial values are set to $x_{0,1}(0) = 1$, $y_{0,1}(0) = 1$, $x_{1,1}(0) = \omega^{-1}$, $y_{1,1}(0) = 1$ and zero for other components. For the numerical computation, the stepsize was set to $h = 0.03$.

Chapter 4

Parallelism in energy-preserving integrators

Contents of Chapter 4 are not publicised, because this chapter is a work of joint authorship and the publication is not approved by a co-author.

Part II

Geometric numerical integration methods for PDEs

Chapter 5

Preliminaries: existing methods and our motivation

In Part II, we discuss geometric numerical integration methods for PDEs. In Section 5.1, some classes of PDEs and the associated properties are summarised. In Sections 5.2, 5.3 and 5.4, symplectic methods, energy-preserving/dissipative methods and multi-symplectic methods are reviewed. These methods are explained by assuming the smoothness and uniqueness of example PDEs. Motivation of the subsequent chapters is explained in Section 5.5.

5.1 Classification of PDEs and their geometric properties

We introduce three classes of PDEs and summarise the associated geometric properties. Note that each class is not separated, and some PDEs belong to all three classes. See [74, 132, 158] for more details.

5.1.1 Hamiltonian PDEs

In this subsection, we introduce the concept of infinite dimensional Hamiltonian systems, which are called Hamiltonian PDEs. It should be noted that Hamiltonian PDEs are natural generalisations of Poisson systems rather than Hamiltonian systems.

First, we show an abstract definition of Hamiltonian PDEs. Briefly speaking, Hamiltonian PDEs are defined by the following replacements

$$\begin{array}{ccc} \text{Poisson systems} & & \text{Hamiltonian PDEs} \\ y_i(t),\ i=1,\dots,N & \to & u(t,x),\ x\in\mathbb{R} \\ \displaystyle\sum_i & \to & \displaystyle\int_{\mathbb{R}} \mathrm{d}x \\ \text{function } H(y) & \to & \text{functional } \mathcal{H}[u] \\ \dfrac{\partial}{\partial y_i} & \to & \dfrac{\delta}{\delta u}. \end{array}$$

A skew-symmetric differential operator $\mathcal{D}(u)$ is called a Poisson operator if the Poisson bracket defined by

$$\{\mathcal{F},\mathcal{G}\}[u]=\int_{\mathbb{R}}\frac{\delta\mathcal{F}}{\delta u}\mathcal{D}(u)\frac{\delta\mathcal{G}}{\delta u}\,\mathrm{d}x \tag{5.1}$$

satisfies the skew-symmetry

$$\{\mathcal{F},\mathcal{G}\}=-\{\mathcal{G},\mathcal{H}\}$$

and the Jacobi identity

$$\{\{\mathcal{F},\mathcal{G}\},\mathcal{H}\}+\{\{\mathcal{G},\mathcal{H}\},\mathcal{F}\}+\{\{\mathcal{H},\mathcal{F}\},\mathcal{G}\}=0, \tag{5.2}$$

where the variational derivative $\frac{\delta\mathcal{H}}{\delta u}$ is defined by

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathcal{H}[u+\epsilon v]\bigg|_{\epsilon=0}=\int_{\mathbb{R}}\frac{\delta\mathcal{H}}{\delta u}v\,\mathrm{d}x.$$

Note that in the discussion here the space $\mathbb{R}$ can be replaced with other spaces such as the torus $\mathbb{T}$. Given a Hamiltonian $\mathcal{H}[u]$, let us consider the motion governed by

$$\frac{\partial}{\partial t}\mathcal{F}[u(t,x)]=\{\mathcal{F},\mathcal{H}\}[u(t,x)].$$

By taking $\mathcal{F}=u$, we obtain a PDE

$$u_t=\mathcal{D}(u)\frac{\delta\mathcal{H}}{\delta u},$$

which is called a Hamiltonian PDE.

Next, we consider two concrete examples.

Example 5.1 (The semi-linear wave equation)**.** The semi-linear wave equation

$$q_{tt}=q_{xx}-f'(q),\qquad x\in\mathbb{T} \tag{5.3}$$

is the simplest case in the sense that this equation can be regarded as an extension of Hamiltonian systems. This equation preserves the symplectic form

$$\Omega=\int_{\mathbb{T}}\mathrm{d}q\wedge\mathrm{d}p\,\mathrm{d}x$$

and the Hamiltonian

$$\mathcal{H}[u]=\int_{\mathbb{T}}\left(\frac{1}{2}p^2+\frac{1}{2}q_x^2+f(q)\right)\mathrm{d}x, \tag{5.4}$$

where $p=q_t$ and $u=(q,p)^\top$. The corresponding Poisson bracket is defined by

$$\{\mathcal{F},\mathcal{G}\}=\int_{\mathbb{T}}\left(\frac{\delta\mathcal{F}}{\delta q}\frac{\delta\mathcal{G}}{\delta p}-\frac{\delta\mathcal{F}}{\delta p}\frac{\delta\mathcal{G}}{\delta q}\right)\mathrm{d}x.$$

This bracket is surely of the form (5.1) with

$$\frac{\delta\mathcal{H}}{\delta u}=\left(\frac{\delta\mathcal{H}}{\delta q},\frac{\delta\mathcal{H}}{\delta p}\right)^\top,\qquad \mathcal{D}=\begin{pmatrix}0&1\\-1&0\end{pmatrix}.$$

Example 5.2 (The Korteweg–de Vries (KdV) equation)**.** The KdV equation

$$u_t=\partial_x\left(\frac{1}{2}u^2+u_x^2\right),\qquad x\in\mathbb{T} \tag{5.5}$$

is a typical example of non-canonical cases. This equation can be rewritten as a variational form

$$u_t=\partial_x\frac{\delta\mathcal{H}}{\delta u},\qquad \mathcal{H}[u]=\int_{\mathbb{T}}\left(\frac{1}{6}u^3-\frac{1}{2}u_x^2\right)\mathrm{d}x,$$

which suggests the Poisson bracket

$$\{\mathcal{F},\mathcal{G}\} = \int_{\mathbb{T}} \frac{\delta \mathcal{F}}{\delta u} \partial_x \frac{\delta \mathcal{G}}{\delta u}\, \mathrm{d}x.$$

The skew-symmetry and Jacobi identity are easily verified. The KdV equation also has another Hamiltonian form

$$u_t = \left(\frac{1}{3}(u\partial_x + \partial_x u) + \partial_x^2\right)\frac{\delta \tilde{H}}{\delta u}, \qquad \tilde{H}[u] = \int_{\mathbb{T}} \frac{u^2}{2}\, \mathrm{d}x.$$

In general, if a PDE has two independent Hamiltonian structures, i.e., bi-Hamiltonian structure, the equation also has infinitely many Hamiltonian structures, which indicates that there are infinitely many invariants. Such a PDE is said to be completely integrable [158, Section 7.3].

5.1.2 Variational PDEs

We consider PDEs which can be written as the variational form

$$u_t = \mathcal{D}(u)\frac{\delta \mathcal{H}}{\delta u},$$

where $\mathcal{D}(u)$ is a skew-symmetric or negative semi-definite differential operator. Hamiltonian PDEs are typical examples of this class.

Let us first consider the case where $\mathcal{D}(u)$ is a skew-symmetric operator with respect to the L^2 inner product. In this case, we also denote the operator by $\mathcal{S}(u)$. Under appropriate boundary conditions with a domain such as $[0, L] \subset \mathbb{R}$, the energy $\mathcal{H}$ is constant along the solution

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H} = \int_0^L \frac{\delta \mathcal{H}}{\delta u} u_t\, \mathrm{d}x = \int_0^L \frac{\delta \mathcal{H}}{\delta u}\mathcal{S}(u)\frac{\delta \mathcal{H}}{\delta u}\, \mathrm{d}x = 0.$$

This property is called the energy-preservation or energy-conservation. We often call $\mathcal{H}$ the energy instead of Hamiltonian. Note that there exist skew-symmetric operators which do not satisfy the Jacobi identity (5.2).

If $\mathcal{D}(u)$ is negative semi-definite (in this case $\mathcal{D}(u)$ is also denoted by $\mathcal{N}(u)$), the equation is dissipative in the sense that

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H} = \int_0^L \frac{\delta \mathcal{H}}{\delta u} u_t\, \mathrm{d}x = \int_0^L \frac{\delta \mathcal{H}}{\delta u}\mathcal{N}(u)\frac{\delta \mathcal{H}}{\delta u}\, \mathrm{d}x \le 0.$$

In this case, $\mathcal{H}$ is called an energy or a Lyapunov function.

Example 5.3 (The Cahn–Hilliard equation)**.** The Cahn–Hilliard equation

$$u_t = \partial_x^2\left(\alpha u + \gamma u^3 + \beta u_{xx}\right) \tag{5.6}$$

with real parameters α, β, γ is a model equation of phase separation. This equation can be written in the variational form

$$u_t = \partial_x^2 \frac{\delta \mathcal{H}}{\delta u}, \qquad \mathcal{H} = \int_0^L \left(\frac{\alpha}{2}u^2 + \frac{\gamma}{4}u^4 - \frac{\beta}{2}u_x^2\right)\mathrm{d}x,$$

with the negative semi-definite operator ∂_x^2. The solution of the Cahn–Hilliard equation is energy-dissipative under the periodic boundary condition. We here show the solution is also dissipative under the boundary condition $u_x|_{x=0,L} = u_{xxx}|_{x=0,L} = 0$. Using the notation $\mathcal{H}[u] = \int_0^L H(u, u_x)\,\mathrm{d}x$, we have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H} &= \int_0^L \left(\frac{\partial H}{\partial u}u_t + \frac{\partial H}{\partial u_x}u_{tx}\right)\mathrm{d}x = \int_0^L \left(\frac{\partial H}{\partial u} - \partial_x\frac{\partial H}{\partial u_x}\right)u_t\,\mathrm{d}x + \left[\frac{\partial H}{\partial u_x}u_t\right]_0^L \\ &= \int_0^L \frac{\delta\mathcal{H}}{\delta u}\partial_x^2\frac{\delta\mathcal{H}}{\delta u}\,\mathrm{d}x = -\int_0^L \left(\partial_x\frac{\delta\mathcal{H}}{\delta u}\right)^2\mathrm{d}x + \left[\frac{\delta\mathcal{H}}{\delta u}\partial_x\frac{\delta\mathcal{H}}{\delta u}\right]_0^L \le 0.\end{aligned}$$

The two boundary terms in the above calculation vanish due to the boundary condition.

5.1.3 Multi-symplectic PDEs

Variational formulations treat the time variable t and space variable x differently. We here summarise a multi-symplectic formulation which treats the two variables equally. A PDE is said to be multi-symplectic, if it can be written as a system of first order equations

$$Mz_t + Kz_x = \nabla_z S(z), \tag{5.7}$$

where $z \in \mathbb{R}^d$ is a vector of state variables including u itself, M, $K \in \mathbb{R}^{d\times d}$ are constant skew-symmetric matrices, and S is a smooth function of z. Note that while the Hamiltonian/variational structures are classical concepts, the multi-symplecticity, which was formulated in 1990s, is a relatively new concept. The variational equation associated with (5.7) is

$$M\mathrm{d}z_t + K\mathrm{d}z_x = S_{zz}\mathrm{d}z,$$

where S_{zz} denotes the Hessian of $S(z)$.

Theorem 5.1 (Multi-symplecticity [18, 19])**.** Let

$$\omega = \mathrm{d}z \wedge M\mathrm{d}z, \qquad \kappa = \mathrm{d}z \wedge K\mathrm{d}z.$$

Then they satisfy the multi-symplectic conservation law

$$\partial_t\omega + \partial_x\kappa = 0.$$

Proof. Because of the skew-symmetry of M, K and symmetry of S_{zz}, it follows that

$$\begin{aligned}\omega_t &= \mathrm{d}z_t \wedge M\mathrm{d}z + \mathrm{d}z \wedge M\mathrm{d}z_t \\ &= -M\mathrm{d}z_t \wedge \mathrm{d}z + \mathrm{d}z \wedge M\mathrm{d}z_t \\ &= -(S_{zz}\mathrm{d}z - K\mathrm{d}z_x) \wedge \mathrm{d}z + \mathrm{d}z \wedge (S_{zz}\mathrm{d}z - K\mathrm{d}z_x) \\ &= -(\mathrm{d}z_x \wedge K\mathrm{d}z + \mathrm{d}z \wedge K\mathrm{d}z_x) = -(\mathrm{d}z \wedge K\mathrm{d}z)_x = -\kappa_x.\end{aligned}$$

□

Theorem 5.2 (Local conservation laws [18, 19])**.** A multi-symplectic PDE has local energy and momentum conservation laws

$$\partial_t E(z) + \partial_x F(z) = 0, \qquad \partial_t I(z) + \partial_x G(z) = 0$$

with the density functions

$$\begin{aligned}E(z) &= S(z) - \frac{1}{2}z_x^\top K^\top z, & F(z) &= \frac{1}{2}z_t^\top K^\top z, \\ G(z) &= S(z) - \frac{1}{2}z_t^\top M^\top z, & I(z) &= \frac{1}{2}z_x^\top M^\top z.\end{aligned}$$

Proof. We show the energy conservation law.

$$\begin{aligned}\partial_t E(z) &= z_t^\top \nabla_z S(z) - \frac{1}{2}(z_x^\top K^\top z)_t = z_t^\top K z_x - \frac{1}{2}(z_x^\top K^\top z)_t \\ &= z_t^\top K z_x - \frac{1}{2} z_{xt}^\top K^\top z - \frac{1}{2} z_x^\top K^\top z_t = -\frac{1}{2}(z_t^\top K^\top z_x + z_{xt}^\top K^\top z) = -\frac{1}{2}(z_t^\top K^\top z)_x = -\partial_x F(z).\end{aligned}$$

The momentum conservation law can be proved in the same manner. □

Integrating the local conservation laws over the spacial domain immediately leads to the global conservation laws.

Theorem 5.3 (Global conservation laws [18, 19])**.** Assume that the boundary conditions are set such that $[F(z)]_0^L = [G(z)]_0^L = 0$. Then, a multi-symplectic PDE has global energy and momentum conservation laws

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{E}(z) = 0, \qquad \mathcal{E}(z) = \int_0^L E(z)\,\mathrm{d}x, \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathcal{I}(z) = 0, \qquad \mathcal{I}(z) = \int_0^L I(z)\,\mathrm{d}x.\end{aligned}$$

Example 5.4 (The KdV equation (5.5))**.** The KdV equation (5.5) has the multi-symplectic structure

$$\underbrace{\begin{pmatrix} 0 & -\frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{M} z_t + \underbrace{\begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}}_{K} z_x = \nabla_z S(z), \tag{5.8}$$

where $z = (\phi, u, v, w)^\top$ and $S(z) = uw + u^3/6 + v^2/2$ [5, 216]. This formulation can be written in the componentwise fashion

$$\begin{aligned}-\frac{1}{2}u_t - w_x &= 0, \\ \frac{1}{2}\phi_t + v_x &= w + \frac{u^2}{2}, \\ -u_x &= v, \\ \phi_x &= u.\end{aligned}$$

The density functions are calculated to be

$$E(z) = \frac{u^3}{6} - \frac{u_x^2}{2} + \frac{(\phi w)_x - (uv)_x}{2}, \qquad I(z) = \frac{\phi_x u - u_x \phi}{4}.$$

Hence, under the periodic boundary condition, we have global conservation laws

$$\mathcal{E}(z) = \int_{\mathbb{T}} \left(\frac{u^3}{6} - \frac{u_x^2}{2} \right) \mathrm{d}x, \qquad \mathcal{I}(z) = \int_{\mathbb{T}} \frac{u^2}{2}\,\mathrm{d}x.$$

5.2 Symplectic methods

The main idea of symplectic methods for Hamiltonian PDEs is to first discretise the equation in space variables such that the resulting system of finite dimensional equations can be seen as Poisson systems, and then apply a symplectic time integration method for the time variable. We illustrate the space discretisation part for the semi-linear wave equation and KdV equation, as our working examples, The following examples are based on [20].

Example 5.5 (The semi-linear wave equation (5.3))**.** We consider a simple approximation to the Hamiltonian (5.4)

$$\mathcal{H}_{\mathrm{d}}(u)=\sum_{k=0}^{N-1}\Delta x\Big(\frac{1}{2}p_k^2+\frac{1}{2}(\delta_x^+q_k)^2+f(q_k)\Big),$$

where $u=(q_0,\dots,q_{N-1},p_0,\dots,p_{N-1})^\top$. Here numerical solutions of $q(t,\cdot)$ on $[0,L]$ with the periodic boundary condition are denoted by $q_k(t)\approx q(t,k\Delta x)$ with $\Delta x=L/N$. A discrete version of the periodic boundary condition is defined by $q_k=q_{k+N}$. Similar notation is also used for p. Let us also define a finite dimensional Poisson bracket by

$$\begin{aligned}\{\mathcal{F}_{\mathrm{d}},\mathcal{G}_{\mathrm{d}}\}&=\frac{1}{\Delta x}\sum_{k=0}^{N-1}\Big(\frac{\partial\mathcal{F}_{\mathrm{d}}}{\partial q_k}\frac{\partial\mathcal{G}_{\mathrm{d}}}{\partial p_k}-\frac{\partial\mathcal{F}_{\mathrm{d}}}{\partial p_k}\frac{\partial\mathcal{G}_{\mathrm{d}}}{\partial q_k}\Big)\\&=\sum_{k=0}^{N-1}\Delta x\Big(\Big(\frac{1}{\Delta x}\frac{\partial\mathcal{F}_{\mathrm{d}}}{\partial q_k}\Big)\Big(\frac{1}{\Delta x}\frac{\partial\mathcal{G}_{\mathrm{d}}}{\partial p_k}\Big)-\Big(\frac{1}{\Delta x}\frac{\partial\mathcal{F}_{\mathrm{d}}}{\partial p_k}\Big)\Big(\frac{1}{\Delta x}\frac{\partial\mathcal{G}_{\mathrm{d}}}{\partial q_k}\Big)\Big).\end{aligned}$$

Then, the finite dimensional equations are given by

$$\dot{u}=\{u,\mathcal{H}_{\mathrm{d}}\},$$

from which we obtain a more concrete form

$$\ddot{q}_k=\delta_x^{\langle2\rangle}q_k-f'(q_k).$$

Example 5.6 (The KdV equation (5.5))**.** Let us define discrete Hamiltonian and Poisson bracket by

$$\mathcal{H}_{\mathrm{d}}(u)=\sum_{k=0}^{N-1}\Delta x\Big(\frac{1}{6}u_k^3-\frac{1}{2}(\delta_x^+u_k)^2\Big)\tag{5.9}$$

and

$$\{\mathcal{F}_{\mathrm{d}},\mathcal{G}_{\mathrm{d}}\}=\frac{1}{\Delta x}\sum_{k=0}^{N-1}\big((\partial_{u_k}\mathcal{F}_{\mathrm{d}})\delta_x^{\langle1\rangle}(\partial_{u_k}\mathcal{G}_{\mathrm{d}})\big).$$

It is easy to check that this bracket satisfies the Jacobi identity (5.2). Then the resulting semi-discrete scheme reads

$$\dot{u}_k=\{u_k,\mathcal{H}_{\mathrm{d}}\}=\delta_x^{\langle1\rangle}\Big(\frac{1}{2}u_k^2+\delta_x^{\langle2\rangle}u_k\Big).$$

5.3 Discrete variational derivative method (energy-preserving/dissipative methods)

5.3.1 Idea of the discrete variational derivative method

In this section, we summarise the so called discrete variational derivative (DVD) method, which was first proposed by Furihata [80, 81] (see also Furihata–Mori [83]), for variational PDEs. The DVD method consists of special spatial discretisation and energy-preserving/dissipative time discretisation such as the discrete gradient method.

Spatial discretisation should be done in such a way that the resulting semi-discrete scheme, i.e., a system of ODEs, has a certain energy-preservation/dissipation property. The main idea of the DVD method is to mimic the proof of the energy-preservation/dissipation property of variational PDEs in a discrete setting.

Below, we summarise the procedure of the DVD method in an abstract form, without focusing on the boundary condition. Then we give some concrete examples.

Discretisation in space

The procedure is summarised as follows.

1. Define a discrete version of the energy $\mathcal{H}_{\mathrm{d}}(u)$ of the form
$$\mathcal{H}_{\mathrm{d}}(u) = \sum_{k=0}^{N-1} \Delta x H_k(u)$$
with functions H_k corresponding to H, where $u = (u_0, \dots, u_{N-1})^\top$.

2. Define a discrete version of the variational derivative $\frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k}$ $(k = 0, \dots, N-1)$, so that it satisfies the chain rule
$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H}_{\mathrm{d}}(u) = \sum_{k=0}^{N-1} \Delta x \frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k}\dot{u}_k.$$

3. Define a skew-symmetric (or negative semi-definite) difference operator $\mathcal{S}_{\mathrm{d}}$ (or $\mathcal{N}_{\mathrm{d}}$) corresponding to the skew-symmetric (or negative semi-definite) differential operator $\mathcal{S}$ (or $\mathcal{N}$).

4. Define a semi-discrete scheme as follows
$$\dot{u}_k = \mathcal{S}_{\mathrm{d}} \frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k}.$$

The semi-discrete scheme preserves the energy
$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H}_{\mathrm{d}}(u) = \sum_{k=0}^{N-1} \Delta x \frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k}\dot{u}_k = \sum_{k=0}^{N-1} \Delta x \frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k}\mathcal{D}_{\mathrm{d}}\frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta u_k} = 0.$$

Discretisation in time

The procedure is summarised as follows.

1. Define a discrete variational derivative $\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}$ $(k=1,\dots,N-1)$, so that it satisfies the *discrete chain rule*

$$\frac{1}{\Delta t}\big(\mathcal{H}_\mathrm{d}(u^{n+1})-\mathcal{H}_\mathrm{d}(u^n)\big)=\sum_{k=0}^{N-1}\Delta x\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}\frac{u_k^{n+1}-u_k^n}{\Delta t}.$$

   In principle, the discrete variational derivative is automatically found by applying the discrete gradient method.

2. Define a fully-discrete scheme as follows

$$\frac{u^{n+1}-u^n}{\Delta t}=\mathcal{S}_\mathrm{d}\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}.$$

The fully-discrete scheme still preserves the energy

$$\frac{1}{\Delta t}\big(\mathcal{H}(u^{n+1})-\mathcal{H}(u^n)\big)=\sum_{k=0}^{N-1}\Delta x\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}\frac{u^{n+1}-u^n}{\Delta t}=\sum_{k=0}^{N-1}\Delta x\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}\mathcal{S}_\mathrm{d}\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}=0.$$

Remark 5.1. In the original DVD method, the space and time variables were discretised simultaneously. However, each part has recently developed separately, and thus it is convenient to treat each variable in tern, as pointed out in [45].

The following examples are based on [80].

Example 5.7 (Energy-preserving finite difference scheme for the KdV equation)**.** We derive an energy-preserving finite difference scheme for the KdV equation (5.5).

First, we define a discrete Hamiltonian by (5.9), and calculate its time derivative

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H}_\mathrm{d}(u)=\sum_{k=0}^{N-1}\Delta x\left(\frac{1}{2}u_k^2\dot{u}_k-(\delta_x^+u_k)(\delta_x^+\dot{u}_k)\right)=\sum_{k=0}^{N-1}\Delta x\left(\frac{1}{2}u_k^2+\delta_x^{\langle 2\rangle}u_k\right)\dot{u}_k.$$

Thus it is natural to define a discrete version of the variational derivative by

$$\frac{\delta \mathcal{H}_\mathrm{d}}{\delta u_k}=\frac{1}{2}u_k^2+\delta_x^{\langle 2\rangle}u_k.$$

Using a skew-symmetric difference operator $\delta_x^{\langle 1\rangle}$, we define a semi-discrete scheme

$$\dot{u}_k=\delta_x^{\langle 1\rangle}\frac{\delta \mathcal{H}_\mathrm{d}}{\delta u_k}.$$

Next, we calculate the discrete chain rule

$$\frac{1}{\Delta t}\big(\mathcal{H}_\mathrm{d}(u^{n+1})-\mathcal{H}_\mathrm{d}(u^n)\big)=\sum_{k=0}^{N-1}\Delta x\left(\frac{(u_k^{n+1})^2+(u_k^{n+1})(u_k^n)+(u_k^n)^2}{6}+\delta_x^{\langle 1\rangle}\left(\frac{u_k^{n+1}+u_k^n}{2}\right)\right)\frac{u_k^{n+1}-u_k^n}{\Delta t}.$$

Based on this calculation, we define the fully-discrete scheme

$$\frac{u^{n+1}-u^n}{\Delta t}=\delta_x^{\langle 1\rangle}\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}$$

with the discrete variational derivative

$$\frac{\delta \mathcal{H}_\mathrm{d}}{\delta(u^{n+1},u^n)_k}=\frac{(u_k^{n+1})^2+(u_k^{n+1})(u_k^n)+(u_k^n)^2}{6}+\delta_x^{\langle 1\rangle}\left(\frac{u_k^{n+1}+u_k^n}{2}\right).$$

Example 5.8 (Energy-dissipative finite difference scheme for the Cahn–Hilliard equation)**.** We derive an energy-dissipative finite difference scheme for the Cahn–Hilliard equation (5.6). In contrast to the KdV case, the boundary condition should be treated more carefully.

First, we define a discrete version of the energy by

$$\mathcal{H}_{\mathrm{d}}(u)=\sum_{k=0}^{N}{}''\Delta x\left(\frac{\alpha}{2}(u_k)^2+\frac{\gamma}{4}(u_k)^4-\frac{\beta}{2}\frac{(\delta_x^+u_k)^2+(\delta_x^-u_k)^2}{2}\right),$$

where $\sum_{k=0}^{N}{}''\Delta x(\cdot)$ denotes the trapezoidal rule:

$$\sum_{k=0}^{N}{}''\Delta x f_k=\Delta x\left(\frac{1}{2}f_0+\sum_{k=1}^{N-1}f_k+\frac{1}{2}f_N\right).$$

Here, numerical solutions of $u(t,\cdot)$ on $[0,L]$ are denoted by $u_k(t)\approx u(t,k\Delta x)$ $(k=0,\dots,N)$ with $\Delta x=L/N$. We also define a discrete version of the boundary condition $u_x|_{x=0,L}=u_{xxx}|_{x=0,L}=0$ by

$$\delta_x^{\langle 1\rangle}u_k\big|_{k=0,N}=\delta_x^{\langle 3\rangle}u_k\big|_{k=0,N}=0\qquad\big(\delta_x^{\langle 3\rangle}:=\delta_x^{\langle 2\rangle}\delta_x^{\langle 1\rangle}\big).$$

The boundary conditions for $\delta_x^{\langle 1\rangle}u_k$ are defined so that the boundary terms in the following calculation are eliminated.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H}_{\mathrm{d}}(u)&=\sum_{k=0}^{N}{}''\Delta x\left(\alpha u_k\dot{u}_k+\gamma(u_k)^3\dot{u}_k-\beta\frac{(\delta_x^+u_k)(\delta_x^+\dot{u}_k)+(\delta_x^-u_k)(\delta_x^-\dot{u}_k)}{2}\right)\\&=\sum_{k=0}^{N}{}''\Delta x\big(\alpha u_k+\gamma(u_k)^3+\beta\delta_x^{\langle 2\rangle}u_k\big)\dot{u}_k-\beta\left[\frac{2(\delta_x^{\langle 1\rangle}u_k)\dot{u}_k+(\delta_x^+u_k)\dot{u}_{k+1}+(\delta_x^-u_k)\dot{u}_{k-1}}{4}\right]_{k=0}^{N}.\end{aligned}$$

We then define a semi-discrete scheme by

$$\dot{u}_k=\delta_x^{\langle 2\rangle}\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k},\qquad\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}=\alpha u_k+\gamma(u_k)^3+\beta\delta_x^{\langle 2\rangle}u_k.$$

The boundary conditions for $\delta_x^{\langle 3\rangle}u_k$ are defined so that boundary terms in the following calculation are eliminated, and the dissipation property follows.

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{H}_{\mathrm{d}}(u)\\&=\sum_{k=0}^{N}{}''\Delta x\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\dot{u}_k=\sum_{k=0}^{N}{}''\Delta x\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\delta_x^{\langle 2\rangle}\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\\&=-\frac{1}{2}\sum_{k=0}^{N}{}''\Delta x\left(\left(\delta_x^+\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\right)^2+\left(\delta_x^-\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\right)^2\right)+\frac{1}{4}\left[2\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\delta_x^{\langle 1\rangle}\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}+\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_{k-1}}\delta_x^-\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}+\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_{k+1}}\delta_x^+\frac{\delta\mathcal{H}_{\mathrm{d}}}{\delta u_k}\right]_0^N\\&\le 0.\end{aligned}$$

Since it follows that

$$\begin{aligned}&\frac{1}{\Delta t}\big(\mathcal{H}_{\mathrm{d}}(u^{n+1})-\mathcal{H}_{\mathrm{d}}(u^n)\big)\\&=\sum_{k=0}^{N}{}''\Delta x\left(\alpha\frac{u_k^{n+1}+u_k^n}{2}+\gamma\frac{(u_k^{n+1})^3+(u_k^{n+1})^2u_k^n+u_k^{n+1}(u_k^n)^2+(u_k^n)^3}{4}-\beta\delta_x^{\langle 2\rangle}\frac{u_k^{n+1}+u_k^n}{2}\right),\end{aligned}$$

we define the fully-discrete scheme

$$\frac{u^{n+1}-u^n}{\Delta t} = \delta_x^{\langle 2\rangle} \frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta (u^{n+1}, u^n)_k}$$

with the discrete variational derivative

$$\frac{\delta \mathcal{H}_{\mathrm{d}}}{\delta (u^{n+1}, u^n)_k} = \alpha \frac{u_k^{n+1}+u_k^n}{2} + \gamma \frac{(u_k^{n+1})^3 + (u_k^{n+1})^2 u_k^n + u_k^{n+1}(u_k^n)^2 + (u_k^n)^3}{4} - \beta \delta_x^{\langle 2\rangle} \frac{u_k^{n+1}+u_k^n}{2}.$$

5.3.2 Extensions of the DVD method

The DVD method has been extended in various ways. Several extensions and applications of the DVD method in 2000s are reviewed in Furihata–Matsuo [82].

Linearisation

The DVD method often produces qualitatively nice numerical solutions. However, a heavy computational cost is inevitable, which is problematic especially in multidimensional problems. This drawback is due to the fact that DVD schemes are implicit and furthermore nonlinear if the target PDE is nonlinear. To avoid the heavy computation, linearly implicit methods have been developed [61, 137]. The idea was already summarised in Remark 2.5, and in most cases the derived schemes actually work very well.

However, there remains a big issue: the derived schemes are sometimes unstable, because the definition of the energy in the linearly implicit methods is different from that in the standard methods. There is no systematic strategy, at the present time, for defining a modified energy so that the corresponding scheme is always stable. For limited examples, an idea for solving this problem was recently proposed by Matsuo–Furihata [138].

Extensions to complicated PDEs

The DVD method originally targeted conservative PDEs of the form

$$u_t = \partial_x^{2s+1} \frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, u_x),$$

and dissipative PDEs of the form

$$u_t = (-1)^{s+1} \partial_x^{2s} \frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, u_x).$$

Then the method has been extended to complex-valued PDEs [137] and second-order PDEs [134], and applied to a lot of PDEs (see [82]). Since around 2010, the method has also been extended/applied to nonlocal PDEs [150, 153, 154, 181, 210] (see also Cohen–Raynaud [57]).

Extensions to nonuniform meshes

The application of the DVD method to spatial discretisation has been restricted to uniform meshes, which requires rectangular domains. This restriction is problematic especially in multidimensional problems, which must frequently be solved in nonuniform meshes. Furthermore, even in one-dimensional cases, nonuniform meshes are often preferred when the solution exhibit locally complicated behaviour. For this reason, several researchers have extended the DVD method to nonuniform meshes. Yaguchi et al. extended the DVD method to nonuniform meshes by either the mapping method [209] or discrete differential forms [211]. Matsuo gave another solution by extending the DVD method to Galerkin framework [135].

5.4 Multi-symplectic methods

In this section, we discuss numerical discretisation methods which preserve the multi-symplectic conservation law (Theorem 5.1). In contrast to symplectic methods and energy-preserving/dissipative methods which treat the space and time variables differently, multi-symplectic methods discretise both variables on equal footing. The main idea of multi-symplectic methods is to apply symplectic discretisation methods also to the space variable. Below, we explain how such an application is done through two specific schemes.

Euler box scheme

Recall that the symplectic Euler method for Hamiltonian systems discretises the variables of position and momenta in a different manner. A similar idea can be applied to multi-symplectic PDEs. Let us introduce a splitting of two matrices M and K, i.e., $M = M_+ + M_-$ and $K = K_+ + K_-$ so that $M_+^\top = -M_-$ and $K_+^\top = -K_-$. The so called Euler box scheme reads

$$M_+\delta_t^+ z_k^n + M_-\delta_t^- z_k^n + K_+\delta_x^+ z_k^n + K_-\delta_x^- z_k^n = \nabla_z S(z_k^n). \tag{5.10}$$

Theorem 5.4 (Moore–Reich [157])**.** The Euler box scheme (5.10) satisfies the discrete multi-symplectic conservation law

$$\delta_t^+ \omega_k^n + \delta_x^+ \kappa_k^n = 0,$$

where

$$\omega_k^n = \mathrm{d}z_k^{n-1} \wedge M_+ \mathrm{d}z_k^n, \qquad \kappa_k^n = \mathrm{d}z_{k-1}^n \wedge K_+ \mathrm{d}z_k^n.$$

Proof. The variational equation associated with (5.10) is

$$M_+\delta_t^+ \mathrm{d}z_k^n + M_-\delta_t^- \mathrm{d}z_k^n + K_+\delta_x^+ \mathrm{d}z_k^n + K_-\delta_x^- \mathrm{d}z_k^n = S_{zz}\mathrm{d}z_k^n.$$

Taking the wedge product with $\mathrm{d}z_k^n$ yields

$$\mathrm{d}z_k^n \wedge \left(M_+\delta_t^+ \mathrm{d}z_k^n + M_-\delta_t^- \mathrm{d}z_k^n\right) + \mathrm{d}z_k^n \wedge \left(K_+\delta_x^+ \mathrm{d}z_k^n + K_-\delta_x^- \mathrm{d}z_k^n\right) = 0$$

due to the symmetry of S_{zz}. Expanding the first term, we have

$$\mathrm{d}z_k^n \wedge \left(M_+\delta_t^+ \mathrm{d}z_k^n + M_-\delta_t^- \mathrm{d}z_k^n\right) = \mathrm{d}z_k^n \wedge M_+\delta_t^+ \mathrm{d}z_k^n + \delta_t^- \mathrm{d}z_k^n \wedge M_+ \mathrm{d}z_k^n = \delta_t^+\left(\mathrm{d}z_k^{n-1} \wedge M_+ \mathrm{d}z_k^n\right) = \delta_t^+ \omega_k^n.$$

Similarly, it is easy to check that the second term becomes

$$\mathrm{d}z_k^n \wedge \left(K_+\delta_x^+ \mathrm{d}z_k^n + K_-\delta_x^- \mathrm{d}z_k^n\right) = \delta_x^+ \kappa_k^n.$$

□

Preissmann box scheme

Recall that the midpoint rule is symplectic. Applying the midpoint rule to both time and space variables in (5.7) leads to the so called Preissmann box scheme or centred box scheme

$$M\delta_t^+ z_{k+1/2}^n + K\delta_x^+ z_k^{n+1/2} = \nabla_z S(z_{k+1/2}^{n+1/2}). \tag{5.11}$$

This scheme was originally introduced by Preissmann in 1961 [166], and has been widely used in hydraulics.

Theorem 5.5 (Bridges–Reich [19])**.** The Preissmann box scheme (5.11) satisfies the discrete multi-symplectic conservation law

$$\delta_t^+ \omega_{k+\frac{1}{2}}^n + \delta_x^+ \kappa_k^{n+\frac{1}{2}} = 0,$$

where

$$\omega_k^n = \mathrm{d}z_k^n \wedge M \mathrm{d}z_k^n, \qquad \kappa_k^n = \mathrm{d}z_k^n \wedge K \mathrm{d}z_k^n.$$

Proof. The variational equation associated with (5.11) is

$$M\delta_t^+ \mathrm{d}z_{k+\frac{1}{2}}^n + K\delta_x^+ \mathrm{d}z_k^{n+\frac{1}{2}} = S_{zz}\mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}}.$$

Taking the wedge product with $\mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}}$ yields

$$\mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}} \wedge M\delta_t^+ \mathrm{d}z_{k+\frac{1}{2}}^n + \mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}} \wedge K\delta_x^+ \mathrm{d}z_k^{n+\frac{1}{2}} = 0$$

due to the symmetry of S_{zz}. Expanding the first term, we have

$$\begin{aligned}\mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}} \wedge M\delta_t^+ \mathrm{d}z_{k+\frac{1}{2}}^n &= \frac{1}{2\Delta t}\left(\mathrm{d}z_{k+\frac{1}{2}}^{n+1} + \mathrm{d}z_{k+\frac{1}{2}}^n\right) \wedge M\left(\mathrm{d}z_{k+\frac{1}{2}}^{n+1} - \mathrm{d}z_{k+\frac{1}{2}}^n\right) \\ &= \frac{1}{2\Delta t}\left(\mathrm{d}z_{k+\frac{1}{2}}^{n+1} \wedge M\mathrm{d}z_{k+\frac{1}{2}}^{n+1} - \mathrm{d}z_{k+\frac{1}{2}}^n \wedge M\mathrm{d}z_{k+\frac{1}{2}}^n\right) = \frac{1}{2}\delta_t^+ \omega_{k+\frac{1}{2}}^n.\end{aligned}$$

For the second term, we can easily check

$$\mathrm{d}z_{k+\frac{1}{2}}^{n+\frac{1}{2}} \wedge K\delta_x^+ \mathrm{d}z_k^{n+\frac{1}{2}} = \frac{1}{2}\delta_x^+ \kappa_k^{n+\frac{1}{2}}$$

in a similar manner. □

Example 5.9 (Preissmann box scheme for the KdV equation [123, Chapter 12])**.** We derive a Preissmann box scheme for the KdV equation (5.5). Applying (5.11) to the multi-symplectic form (5.8), we obtain

$$\begin{aligned}-\frac{1}{2}\delta_t^+ u_{k+1/2}^n - \delta_x^+ w_k^{n+1/2} &= 0, \\ \frac{1}{2}\delta_t^+ \phi_{k+1/2}^n + \delta_x^+ v_k^{n+1/2} &= w_{k+1/2}^{n+1/2} + \frac{\left(u_{k+1/2}^{n+1/2}\right)^2}{2}, \\ -\delta_x^+ u_k^{n+1/2} &= v_{k+1/2}^{n+1/2}, \\ \delta_x^+ \phi_k^{n+1/2} &= u_{k+1/2}^{n+1/2}.\end{aligned}$$

By introducing two matrices

$$A = \frac{1}{2}\begin{pmatrix} 1 & 1 & & & \\ & 1 & 1 & & \\ & & \ddots & \ddots & \\ & & & 1 & 1 \\ 1 & & & & 1 \end{pmatrix} \in \mathbb{R}^{N\times N} \qquad \text{and} \qquad D = \frac{1}{\Delta x}\begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \\ 1 & & & & -1 \end{pmatrix} \in \mathbb{R}^{N\times N}$$

and assuming $\delta_x^+ u_k^0 = v_{k+1/2}^0$ and $\delta_x^+ \phi_k^0 = u_{k+1/2}^0$, the above scheme can be rewritten as a vector form

$$\begin{aligned} -\frac{1}{2}\delta_t^+ Au^n - Dw^{n+1/2} &= 0, \\ \frac{1}{2}\delta_t^+ A\phi^n + Dv^{n+1/2} &= Aw^{n+1/2} + \frac{\left(Au^{n+1/2}\right)^2}{2}, \\ -Du^n &= Av^n, \\ D\phi^n &= Au^n, \end{aligned}$$

where $u^n = (u_0^n, \dots, u_{N-1}^n)^\top$, similar notation is used for other variables, and $\left(Au^{n+1/2}\right)^2$ denotes the componentwise vector product. The matrix A is invertible if N is odd. In this case, eliminating ϕ^n, v^n and w^n, we finally obtain the scheme

$$\delta_t^+ Au^n = DA^{-1}\frac{\left(Au^{n+1/2}\right)^2}{2} + DA^{-1}DA^{-1}Du^{n+1/2}.$$

Backward error analysis

As is the case with the symplectic methods for Hamiltonian systems, the multi-symplectic methods approximate a multi-symplectic PDE by another multi-symplectic PDE. In other words, the solution of a multi-symplectic integrator is an exact solution of the modified multi-symplectic equation. Hence, the errors of the energy and momentum of the multi-symplectic methods are bounded by a constant independently of t. Moore–Reich showed this property by using the backward error analysis [157] (see also [156]). Below, we show the idea of the backward error analysis taking the Euler box scheme as our working example.

We consider the Taylor series of the exact solution $z(t_{n+1}, x_k)$ around (t_n, x_k):

$$z(t_{n+1}, x_k) = z(t_n, x_k) + \Delta t z_t(t_n, x_k) + \frac{\Delta t^2}{2} z_{tt}(t_n, x_k) + \frac{\Delta t^3}{6} z_{ttt}(t_n, x_k) + \mathcal{O}(\Delta t^4).$$

From this, we immediately obtain

$$\delta_t^+ z(t_n, x_k) = z_t(t_n, x_k) + \frac{\Delta t}{2} z_{tt}(t_n, x_k) + \frac{\Delta t^2}{6} z_{ttt}(t_n, x_k) + \mathcal{O}(\Delta t^3).$$

Similarly, it follows that

$$\begin{aligned} \delta_t^- z(t_n, x_k) &= z_t(t_n, x_k) - \frac{\Delta t}{2} z_{tt}(t_n, x_k) + \frac{\Delta t^2}{6} z_{ttt}(t_n, x_k) - \mathcal{O}(\Delta t^3), \\ \delta_x^+ z(t_n, x_k) &= z_x(t_n, x_k) + \frac{\Delta x}{2} z_{xx}(t_n, x_k) + \frac{\Delta x^2}{6} z_{xxx}(t_n, x_k) + \mathcal{O}(\Delta x^3), \\ \delta_x^- z(t_n, x_k) &= z_x(t_n, x_k) - \frac{\Delta x}{2} z_{xx}(t_n, x_k) + \frac{\Delta x^2}{6} z_{xxx}(t_n, x_k) - \mathcal{O}(\Delta x^3). \end{aligned}$$

Substituting these expressions into (5.10), we obtain

$$Mz_t + \frac{\Delta t}{2}(M_+ - M_-)z_{tt} + Kz_x + \frac{\Delta x}{2}(K_+ - K_-)z_{xx} = \nabla_z S(z). \tag{5.12}$$

Here, higher-order terms $\mathcal{O}(\Delta t^2 + \Delta x^2)$ are omitted to simplify the notation, but even though such terms are taken into account, the following discussion still makes sense. The PDE (5.12) is multi-symplectic. Indeed, by letting $L = (M_+ - M_-)$ and $N = (K_+ - K_-)$, the PDE (5.12) is equivalent to

$$\tilde{M}\tilde{z}_t + \tilde{K}\tilde{z}_x = \nabla_{\tilde{z}}\tilde{S}(\tilde{z}),$$

where $\tilde{z} = (z, z_t, z_x)^\top$,

$$\tilde{M} = \begin{pmatrix} M & \frac{\Delta t}{2}L & 0 \\ -\frac{\Delta t}{2}L & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad \tilde{K} = \begin{pmatrix} K & 0 & \frac{\Delta x}{2}N \\ 0 & 0 & 0 \\ -\frac{\Delta x}{2}N & 0 & 0 \end{pmatrix}$$

and

$$S(\tilde{z}) = S(z) - \frac{\Delta}{2} z_t^\top L z_t - \frac{\Delta x}{2} z_x^\top N z_x.$$

Therefore, the Euler box scheme exactly preserves the energy and momentum of the modified multi-symplectic PDE.

Extensions/applications of multi-symplectic method

In addition to the Euler box scheme and Preissmann box scheme, several discretisation methods have been proposed. For example, Reich applied Runge–Kutta methods to both space and time variables [170] (see also Runge–Kutta–Nyström methods [109, 130] and partitioned Runge–Kutta methods [129]). In contrast to energy-preserving/dissipative methods, every discretisation method can be readily applied to a multi-symplectic formulation once the structure is found. Multi-symplectic formulations of a variety of PDEs have been found (see, e.g., [55, 56, 108]).

5.5 Motivation and summary of the subsequent chapters

Chapter 6

Among several recent developments of the discrete variational derivative (DVD) method to nonuniform meshes and high-dimensional problems, we focus on the Galerkin framework proposed by Matsuo [135], which is referred to as the discrete partial derivative (DPD) method. He targeted dissipative PDEs of the form

$$\frac{\partial u}{\partial t} = (-1)^{s+1}\left(\frac{\partial}{\partial x}\right)^{2s} \frac{\delta \mathcal{H}}{\delta u}, \qquad s = 0, 1, 2, \ldots, \tag{5.13}$$

and conservative PDEs

$$\frac{\partial u}{\partial t} = \left(\frac{\partial}{\partial x}\right)^{2s-1} \frac{\delta \mathcal{H}}{\delta u}, \qquad s = 1, 2, 3, \ldots, \tag{5.14}$$

where $\delta\mathcal{H}/\delta u$ is the variational derivative of $H(u, u_x)$ with respect to $u(t, x)$. The DPD method has been applied to several specific PDEs [120, 136, 139]. Various dissipative/conservative schemes proposed in the literature have been identified as the special cases of the method, including the famous Du–Nicolaides scheme for the Cahn–Hilliard equation [69]. These findings demonstrate the comparative success of the DPD method.

However, a major limitation remains in the DPD method. First, the DPD method defines H^1-weak forms with explicit dissipation or conservation properties, and then discretises them appropriately with P1 elements. P1 elements reduce the computational complexity, particularly in two- or three-dimensional problems. However, as the variational structures become more complicated than those in (5.13) and (5.14), finding an appropriate H^1-weak form for PDEs becomes increasingly difficult. Such PDEs can be categorised into two types.

Type 1: PDEs whose energy functional contains higher-order derivatives, i.e., (5.13) or (5.14) with $H = H(u, u_x, u_{xx}, \dots)$. For such PDEs, the energy in H^1 space is not clearly defined. An example of these dissipative PDEs is the Swift–Hohenberg (SH) equation [180]

$$\frac{\partial u}{\partial t} = -\left(-2u + u^3 + 2\frac{\partial^2 u}{\partial x^2} + \frac{\partial^4 u}{\partial x^4}\right), \qquad 0 < x < L, \quad t > 0.$$

This equation belongs to class (5.13) with $s = 0$ and $H(u, u_x, u_{xx}) = -u^2 + u^4/4 - u_x^2 + u_{xx}^2/2$. An example of a conservative PDE is the Kawahara equation (fifth-order KdV-type equation) [117]

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(-\frac{1}{2}u^2 - \alpha\frac{\partial^2 u}{\partial x^2} + \beta\frac{\partial^4 u}{\partial x^4}\right), \qquad 0 < x < L, \quad t > 0. \tag{5.15}$$

This equation belongs to class (5.14) with $s = 1$ and $H(u, u_x, u_{xx}) = -u^3/6 + \alpha u_x^2/2 + \beta u_{xx}^2/2$.

Type 2: PDEs in which a complicated differential operator acts on the variational derivative (specifically, the operator $\mathcal{S}$ in $u_t = \mathcal{S}\frac{\delta \mathcal{H}}{\delta u}$ is complicated). In this case, the dissipation or conservation property arises from the negative semi-definite or skew-symmetric property of the differential operator. When the operator cannot operate on functions in H^1, special treatments are required. Typical examples are the Camassa–Holm equation [41, 42, 79]

$$u_t - u_{xxt} = uu_{xxx} + 2u_x u_{xx} - 3uu_x, \qquad 0 < x < L, \quad t > 0, \tag{5.16}$$

and the Degasperis–Procesi equation [66]

$$u_t - u_{xxt} = uu_{xxx} + 3u_x u_{xx} - 4uu_x, \qquad 0 < x < L, \quad t > 0. \tag{5.17}$$

The Camassa–Holm equation can be written in variational (Hamiltonian) form with $\mathcal{S} = (1 - \partial_x^2)^{-1}(m\partial_x + \partial_x m)(1 - \partial_x^2)^{-1}$ and $H(u, u_x) = -(u^2 + u_x^2)/2$. Similarly, the Degasperis–Procesi equation can be written with $\mathcal{S} = (1 - \partial_x^2)^{-1}\partial_x(4 - \partial_x^2)$ and $H(u) = -u^3/6$.

To our knowledge, no systematic procedure exists for finding dissipative or conservative H^1-weak forms of the above equation types. This complicates the application of the DPD method. One solution is to adopt smoother function spaces; however we would prefer to stick to H^1-weak forms since

- the H^1-formulation can be implemented by computationally inexpensive P1 elements. This advantage is mandatory in multidimensional problems;
- for high-order PDEs with H^1 solutions, such as the Camassa–Holm equation (which has peaked soliton solutions), H^1-formulations are preferable from a theoretical perspective.

With these considerations, we propose a new framework that automatically constructs H^1 schemes for Types 1 and 2 PDEs. Similar to the original DPD method, the proposed method exploits the variational structure of PDEs, but finds the intended schemes rather than the underlying dissipative or conservative H^1-weak forms. This nontrivial approach is rendered possible by the concept of L^2-projection operators. Note that the proposed method is not a superset of the original DPD method, in the sense that the proposed method often derives different schemes from the original DPD method for the simple PDEs (5.13) and (5.14). Moreover, by using the proposed method, we propose a new class of energy-preserving/dissipative methods in discontinuous-Galerkin framework.

Chapter 7

There remain other issues to be settled for energy-preserving/dissipative methods. For example, they assumed *static* grids, and it is not clear at all if it could be incorporated with a dynamic grid adaptation technique. Such a technique is required in practical problems where a localised point (or area) moves as time passes (consider, for example, a moving solitary wave), in order to increase the overall efficiency. Unfortunately, however, it seems that no study has ever succeeded in such a challenge, not only in the context of energy-preserving/dissipative methods, but also in the more general context of geometric numerical integration methods for PDEs, except in very specific studies such as [219]. The reason for this is that such structure-preserving methods usually employ a very sophisticated time stepping for the desired structure-preservation, which generally seems to contradict the concept of grid adaptation.

Motivated by this background, in Chapter 7 we shall show that by a simple idea we can establish an adaptive energy-preserving/dissipative method. This is done by combining the energy-preserving/dissipative method on static nonuniform grids and a grid adaptation technique. We here would like to emphasise that a simple combination of them would destroy the desired properties. The key is to introduce an additional optimisation step, by which the destruction can be avoided.

Remark 5.2. As noted in Section 1.2, Chapter 7 is mainly based on [152]. During the revision period of [152], we noticed a recent, essentially equivalent study [71]. While in [152] the energy-preserving/dissipative Galerkin method and a wavelet-based grid adaptation technique are combined, in [71] the energy-preserving/dissipative finite difference method on nonuniform grids and an equidistribution-based grid adaptation technique are incorporated. In Chapter 7, the equidistribution-based grid adaptation technique is also considered.

Chapter 8

A time-dependent PDE is usually formulated locally, in the sense that it is of the form

$$u_t = f(u, u_x, u_{xx}, \dots).$$

On the other hand, in the last few decades, the study of nonlocal PDEs has become popular in several research fields such as wave theory, PDE theory and integrability. Here lists several, famous examples.

- b-family equation:

  $$(1-\partial_x^2)u_t = -(b+1)uu_x + bu_xu_{xx} + uu_{xxx}.$$

  When $b = 2$ this equation is called the Camassa–Holm equation (5.16). When $b = 3$ this equation is called the Degasperis–Procesi equation (5.17). Both equations describe shallow water waves, and only these equations are completely integrable.

- Ostrovsky equation [159]

  $$u_t + \alpha uu_x - \beta u_{xxx} = \gamma\partial_x^{-1}u,$$

  where α, β and γ are real parameters.

- Hunter–Saxton equation [111]

  $$u_{xxt} + 2u_xu_{xx} + uu_{xxx} = 0. \tag{5.18}$$

- Short-pulse equation [50, 174]

  $$u_{xt} = u + \frac{1}{6}(u^3)_{xx}.$$

For nonlocal PDEs, nonlocal operators, such as $(1-\partial_x^2)^{-2}$, ∂_x^{-1} and ∂_x^{-2}, should be carefully handled, so that the intended geometric structure is kept even after the discretisation. It is strongly hoped to construct a general framework for handling these nonlocal operators. However, since the treatment of nonlocal operators is also related to boundary conditions, structure-preserving discretisations of nonlocal PDEs are not obvious at all in general, and have been considered for individual equation, as it is now.

The Camassa–Holm equation has been intensively studied in the last decade. This equation is usually considered under the periodic boundary conditions. Under the periodic boundary conditions, the operator $(1-\partial_x^2)^{-1}$ is invertible. This property can be kept in the finite difference context, and thus structure-preserving schemes can be derived. Energy-preserving schemes were proposed by Takeya [181] (see also [82, Chapter 4.7] and [153]). Multi-symplectic integrators were proposed by Cohen et al. [56]. The operator $(1-\partial_x^2)^{-n}$ of the modified Camassa–Holm equation was considered in [153].

In the context of structure-preserving numerical methods, the operator ∂_x^{-1} of the Ostrovsky equation was discussed by Yaguchi et al. [210] and Miyatake et al. [154]. For this equation, we usually assume the existence of a potential $\phi = \partial_x^{-1}u$ under the periodic boundary condition. In other words, $\int_{\mathbb{T}} u\,\mathrm{d}x = 0$ is assumed for all $t > 0$. Structure-preserving schemes derived in [154, 210] are also based on this assumption.

In Chapter 8, we will consider the Hunter–Saxton equation which is associated with the operator ∂_x^{-2}. Readers might feel that since the treatment of ∂_x^{-1} was already studied, it is straightforward to handle ∂_x^{-2}. However, it is not easy to derive structure-preserving schemes for the Hunter–Saxton equation, because the existence of the potential is not assumed in general for the equation.

Since its introduction in the seminal paper [111], the Hunter–Saxton equation has been attracting much attention. This is mainly due to its rich mathematical structures: the Hunter–Saxton equation is integrable; it is bi-Hamiltonian; it possess a Lax pair; it does not have global smooth solutions but enjoy two distinct classes of global weak solutions (conservative and dissipative); it can be seen as the geodesic equation of a right-invariant metric on a certain quotient space; etc. (see [17, 112, 113, 125] and references therein). Furthermore, the Hunter–Saxton equation arises as a model for the propagation of weakly nonlinear orientation waves in a nematic liquid crystal [111] and it can be seen as the high frequency limit of another well known and well studied equation, namely the Camassa–Holm equation. Note that More about this last equation can be found, for example, in the work [168], the recent review [106], and references therein.

Furthermore, there are a lot of ongoing research activities on the two extensions of the Hunter–Saxton equation: the modified Hunter–Saxton equation (introduced in [124])

$$u_{xxt} + 2u_x u_{xx} + uu_{xxx} - 2\omega u_x = 0,$$

where $\omega > 0$, and the two-component Hunter–Saxton system (introduced in [200])

$$\begin{aligned} &u_{xxt} + 2u_x u_{xx} + uu_{xxx} - \kappa\rho\rho_x = 0,\\ &\rho_t + (u\rho)_x = 0, \end{aligned}$$

where $\kappa \in \{-1, 1\}$ and $\rho := \rho(x,t)$. These two equations also enjoy many interesting properties. The modified Hunter–Saxton equation is a model for short capillary waves propagating under the action of gravity [126]. An interesting feature of this modified version of the original problem is that it admits (smooth as well as cusped) travelling waves. This is not the case for the original problem (5.18). Moreover, this partial differential equation is also bi-Hamiltonian [126]. The two-component generalisation of the Hunter–Saxton equation is a particular case of the Gurevich–Zybin system which describes the dynamics in a model of non-dissipative dark matter, see [163] and also [128]. As the original equation, this system is integrable; has a Lax pair; is bi-Hamiltonian; it is also the high-frequency limit of the two-component Camassa–Holm equation; has peakon solutions; its flow is equivalent to the geodesic flow on a certain sphere; etc. (see [118, 127, 155, 198, 201] and references therein).

Despite the fact that the above equations are well understood in a more theoretical way, there are not much results on numerical discretisations of these problems. We are aware of the numerical schemes from [105, 206, 207] proposed only for the original Hunter–Saxton equation. The work [105] proves convergence of some discrete finite difference schemes to dissipative solutions of the Hunter–Saxton equation on the half-line. The references [206, 207] analyse local discontinuous Galerkin methods for the Hunter–Saxton equation and in particular, using results from [105], prove convergence of the discretisation scheme to the dissipative solutions. The main goal of this chapter is to present novel numerical discretisations of the Hunter–Saxton equation and of its above two generalisations. The proposed numerical schemes are based on a multi-symplectic, resp. on a variational, formulation of the problems. They are specially designed to preserve two geometric features of the original partial differential equation, namely the multi-symplectic structure and the conservative property of the problem. In the present work, we are not concerned about convergence results.

Chapter 6

A general Galerkin framework with L^2-projection

In this chapter, a general Galerkin framework for automatically deriving energy-preserving/dissipative schemes is proposed. In Section 6.1, we review the original discrete partial derivative (DPD) method and clarify its limitation. We propose a new framework for one-dimensional PDEs in Section 6.2, and then extend it to multidimensional problems in Section 6.3. Furthermore, we combine the new framework with the discontinuous Galerkin methods in Section 6.4.

We use the following notation. The numerical solution is denoted by $u^{(n)} \simeq u(n\Delta t, \cdot)$, where Δt is the time step. For a positive integer j and appropriate domain and boundary conditions, H^j denotes the standard Sobolev space equipped with the norm

$$\|f\|_{H^j} = \left(\|f\|_{L^2}^2 + \sum_{l=1}^{j} \|\partial_x^l f\|_{L^2}^2 \right)^{\frac{1}{2}}.$$

In one-dimensional problems, the interval (domain) is set to $[0, L]$, and the L^2 inner product is defined by $(f, g) = \int_0^L f\, g \,\mathrm{d}x$. S_i and W_i $(i = 1, 2, \dots)$ denote the trial and test spaces, respectively. When imposing periodic boundary conditions, we often use the notation $H^1(\mathbb{T})$ (where $\mathbb{T}$ denotes the torus of length L).

In multidimensional problems, the domain is specified by $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$. The L^2 inner product is defined by $(f, g) = \int_\Omega f\, g \,\mathrm{d}x$ when f and g are scalar-valued functions, and by $(\boldsymbol{f}, \boldsymbol{g}) = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{g} \,\mathrm{d}x$ when $\boldsymbol{f}$ and $\boldsymbol{g}$ are vector-valued functions (the dot signifies $\boldsymbol{f} \cdot \boldsymbol{g} = \boldsymbol{f}^\top \boldsymbol{g}$). $\boldsymbol{L}^2$ and $\boldsymbol{H}^j$ denote $(L^2)^d$ and $(H^j)^d$, respectively. $\Gamma = \partial\Omega$ and $\boldsymbol{n}$ denote the boundary of Ω and the normal vector at the boundary, respectively. The Green theorem

$$\int_\Omega (f \triangle g + \nabla f \cdot \nabla g) \,\mathrm{d}x = \int_\Gamma f \boldsymbol{n} \cdot \nabla g \,\mathrm{d}\Gamma \tag{6.1}$$

is used instead of the integration-by-parts formula.

Later, we shall apply the proposed framework to several PDEs. However, since the aim of such applications is just to support the advantages of the framework: the wide range of applications and the completely automatic procedure, we do not mind whether or not global/local well-posedness of the exact solution for each equation is already verified in the PDE theory.

6.1 Discrete partial derivative method and its limitation

In this section, we introduce the original DPD method [135] and its essential limitations. The DPD method is implemented in three steps, as shown below.

Step 1 Construct an H^1-weak form that explicitly expresses the desired dissipation/conservation property.

Step 2 Spatially discretise the weak form such that the resulting semi-discrete scheme is consistent in some finite-dimensional approximation spaces of H^1 and it retains the dissipation/conservation property.

Step 3 Temporally discretise the semi-discrete scheme such that the desired property is retained (this step is essentially that of the discrete gradient method).

As an example, we demonstrate Steps 1–3 for the dissipative case (5.13) with $s = 0$.

Step 1

First, we define a dissipative H^1-weak form with

$$\frac{\delta \mathcal{H}}{\delta u} = \frac{\partial H}{\partial u} - \frac{\partial}{\partial x}\frac{\partial H}{\partial u_x}$$

in mind.

Weak form 1 (dissipative H^1-weak form for (5.13) when $s = 0$ [135])**.** Suppose that $u(0,\cdot)$ is given in $H^1(0,L)$. We find $u(t,\cdot) \in H^1(0,L)$ such that for any $v \in H^1(0,L)$,

$$(u_t, v) = -\left(\frac{\partial H}{\partial u}, v\right) - \left(\frac{\partial H}{\partial u_x}, v_x\right) + \left[\frac{\partial H}{\partial u_x} v\right]_0^L. \tag{6.2}$$

Proposition 6.1 (Weak form 1: dissipation property [135])**.** Assume that the boundary conditions satisfy

$$\left[\frac{\partial H}{\partial u_x} u_t\right]_0^L = 0, \tag{6.3}$$

and also assume that $u_t(t,\cdot) \in H^1(0,L)$, $\frac{\partial H}{\partial u} \in L^2(0,L)$ and $\frac{\partial H}{\partial u_x} \in L^2(0,L)$. Then the solution of Weak form 1 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,u_x)\,\mathrm{d}x \le 0.$$

Proof.

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,u_x)\,\mathrm{d}x = \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) = -\|u_t\|^2 + \left[\frac{\partial H}{\partial u_x} u_t\right]_0^L \le 0.$$

The first equality is a simple application of the chain rule, while the second follows from (6.2) with $v = u_t \in H^1(0,L)$. The last inequality follows from assumption (6.3). □

Note that the partial derivatives $\partial G/\partial u$ and $\partial G/\partial u_x$ play an important role in constructing the above weak form.

Step 2

In this step, the function space $H^1(0,L)$ in Weak form 1 is replaced by finite-dimensional approximation spaces S_1 and $W_1 \subset H^1(0,L)$. Thus, we obtain the following semi-discrete scheme.

Semi-discrete scheme 1 (semi-discrete dissipative scheme for (5.13) when $s = 0$ [135])**.** Suppose that $u(0,\cdot)$ is given in S_1. We find $u(t,\cdot) \in S_1$ such that for any $v \in W_1$,

$$(u_t, v) = -\left(\frac{\partial H}{\partial u}, v\right) - \left(\frac{\partial H}{\partial u_x}, v_x\right) + \left[\frac{\partial H}{\partial u_x} v\right]_0^L.$$

Proposition 6.2 (Semi-discrete scheme 1: dissipation property [135])**.** Assume that the boundary conditions and the trial and test spaces are set to satisfy $\left[\frac{\partial H}{\partial u_x} u_t\right]_0^L = 0$ and $u_t \in W_1$. Also assume that $\frac{\partial H}{\partial u} \in L^2(0,L)$ and $\frac{\partial H}{\partial u_x} \in L^2(0,L)$. Then the solution of Semi-discrete scheme 1 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_0^L H(u, u_x)\,\mathrm{d}x \le 0.$$

Proof. The proof is similar to that of Proposition 6.1:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_0^L H(u, u_x)\,\mathrm{d}x = \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) = -\|u_t\|^2 + \left[\frac{\partial H}{\partial u_x} u_t\right]_0^L \le 0.$$

The substitution $v = u_t$ is allowed by the assumption $u_t \in W_1$. □

Step 3

To discretise Semi-discrete scheme 1 in time, we introduce discrete partial derivatives.

Definition 6.1 (discrete partial derivatives [135])**.** For two functions $u^{(n+1)}$ and $u^{(n)}$, we call the discrete quantities

$$\frac{\partial H_\mathrm{d}}{\partial (u^{(n+1)}, u^{(n)})} \qquad \text{and} \qquad \frac{\partial H_\mathrm{d}}{\partial (u_x^{(n+1)}, u_x^{(n)})}, \tag{6.4}$$

which correspond to $\partial H/\partial u$ and $\partial H/\partial u_x$, "discrete partial derivatives," if they satisfy the following identity:

$$\begin{aligned}
&\frac{1}{\Delta t} \int_0^L \left(H(u^{(n+1)}, u_x^{(n+1)}) - H(u^{(n)}, u_x^{(n)})\right) \mathrm{d}x \\
&= \left(\frac{\partial H_\mathrm{d}}{\partial (u^{(n+1)}, u^{(n)})}, \frac{u^{(n+1)} - u^{(n)}}{\Delta t}\right) + \left(\frac{\partial H_\mathrm{d}}{\partial (u_x^{(n+1)}, u_x^{(n)})}, \frac{u_x^{(n+1)} - u_x^{(n)}}{\Delta t}\right).
\end{aligned} \tag{6.5}$$

Note that the subscript d just means that we consider discrete versions of the partial derivatives (H_d does not make any sense). Since (6.5) corresponds to the continuous chain rule:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_0^L H(u, u_x)\,\mathrm{d}x = \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right),$$

it is referred to as the discrete chain rule. Calculations of discrete partial derivatives are given in [135]: essentially, these calculations can be done based on the discrete gradient method or more specifically AVF method (see Section 2.5). For example, for $H(u, u_x) = \frac{1}{6}u^3 - \frac{1}{2}u_x^2$ so that $\frac{\partial H}{\partial u} = \frac{1}{2}u^2$ and $\frac{\partial H}{\partial u_x} = -u_x$,

$$\frac{\partial H_\mathrm{d}}{\partial (u^{(n+1)}, u^{(n)})} = \frac{(u^{(n+1)})^2 + u^{(n+1)}u^{(n)} + (u^{(n)})^2}{6} \qquad \text{and} \qquad \frac{\partial H_\mathrm{d}}{\partial (u_x^{(n+1)}, u_x^{(n)})} = -\frac{u^{(n+1)} + u^{(n)}}{2}$$

satisfy the discrete chain rule (6.5).

Let us return to the temporal discretisation of Semi-discrete scheme 1. Replacing the time derivative u_t with $(u^{(n+1)} - u^{(n)})/\Delta t$, and the partial derivatives $\partial H/\partial u$ and $\partial H/\partial u_x$ with the discrete partial derivatives (6.4), we obtain the following fully discrete scheme.

Scheme 1 (dissipative Galerkin scheme for (5.13) when $s=0$ [135])**.** Suppose that $u^{(0)}$ is given in S_1. Find $u^{(n+1)} \in S_1$ $(n=0,1,\dots)$ such that for any $v \in W_1$,

$$\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}, v\right) = -\left(\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)}, u^{(n)})}, v\right) - \left(\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})}, v_x\right) + \left[\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})} v\right]_0^L. \quad (6.6)$$

Theorem 6.1 (Scheme 1: dissipation property [135])**.** Assume that the boundary conditions and the trial and test spaces are set to satisfy

$$\left[\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})}\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right)\right]_0^L = 0 \quad (6.7)$$

and $(u^{(n+1)}-u^{(n)})/\Delta t \in W_1$. Also assume that $\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)},u^{(n)})} \in L^2(0,L)$ and $\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)},u_x^{(n)})} \in L^2(0,L)$. Then the solution of Scheme 1 satisfies

$$\frac{1}{\Delta t}\int_0^L \left(H(u^{(n+1)}, u_x^{(n+1)}) - H(u^{(n)}, u_x^{(n)})\right) \mathrm{d}x \le 0, \qquad n=0,1,2,\dots.$$

Proof.

$$\begin{aligned}
&\frac{1}{\Delta t}\int_0^L \left(H(u^{(n+1)}, u_x^{(n+1)}) - H(u^{(n)}, u_x^{(n)})\right) \mathrm{d}x \\
&= \left(\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)}, u^{(n)})}, \frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right) + \left(\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})}, \frac{u_x^{(n+1)}-u_x^{(n)}}{\Delta t}\right) \\
&= -\left\|\frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right\|^2 + \left[\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})}\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right)\right]_0^L \le 0.
\end{aligned}$$

The first equality is a simple application of the discrete chain rule (6.5), while the second follows from (6.6) with $v = (u^{(n+1)}-u^{(n)})/\Delta t \in W_1$. The last inequality follows from assumption (6.7). □

Note that Steps 2 and 3 are completely automatic. In other words, once a dissipative/conservative H^1-weak form is found, an intended fully discrete Galerkin scheme can be systematically obtained. Dissipative/conservative H^1-weak forms for (5.13) and (5.14) have been published in the literature [135].

Unfortunately, however, finding the desired weak forms of complicated PDEs is less straightforward. The partial derivatives of Type 1 PDEs do not always exist in H^1 space. For Type 2 PDEs, the complicated operator is not easily handled in H^1 space.

6.2 New framework for one-dimensional problems

To overcome the above difficulty, we propose a new framework. The new procedure is summarized as follows (Figure 6.1).

PHASE 1

Step 1′ Construct a *formal* weak form that need not be formulated within H^1 space, but whose dissipation/conservation property can be explicitly obtained by formal calculations. The meaning of "*formal*" will be clarified shortly.

Step 2′ Spatially discretise the *formal* weak form. The resulting semi-discrete scheme should be consistent in some finite-dimensional approximation spaces of H^1 and should retain the dissipation/conservation property. In this step, the L^2-projection operators play an important role.

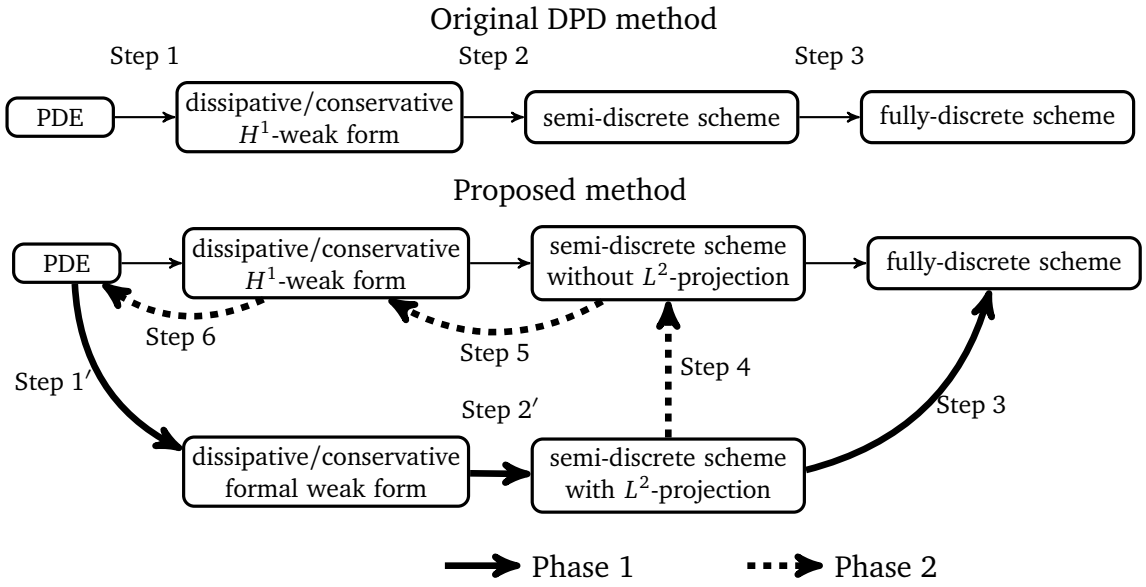


Figure 6.1: Standard versus proposed strategies.

Step 3 Temporally discretise the semi-discrete scheme such that the desired property is retained.

If the theoretical aspects of the schemes (such as convergence behaviour) are of interest, finding the underlying dissipative/conservative H^1-weak forms is assisted by the following steps.

PHASE 2

Step 4 The semi-discrete scheme derived in Step 2′ explicitly uses the L^2-projection operators. In this step, these operators are expanded into a more familiar form.

Step 5 Restore the semi-discrete scheme to a weak form that is consistent in H^1. This step merely rewrites the finite-dimensional approximation spaces to infinite-dimensional subspaces of H^1.

Step 6 Check the relationship between PDE and the obtained H^1-weak form.

The proposed method avoids the need to find obscure proper dissipative/conservative H^1-weak forms. This advantage is conferred by the L^2-projection operators. Note that our approach is completely automatic, except that some degrees of freedom must be specified in Step 2′ (see Remark 6.6).

The L^2-projection operators are introduced in Section 6.2.1. In Section 6.2.2 and Section 6.2.3, the proposed method is defined for Types 1 and 2 PDEs, respectively. Applications of the method are presented in Section 6.2.4.

Remark 6.1. In Propositions 6.1 and 6.2 and Theorem 6.1, we assumed that partial derivatives and discrete partial derivatives are in L^2. In the rest of this chapter, we do not write similar assumptions explicitly, but we promise that they are always assumed.

6.2.1 L^2-projection operators

The L^2-projection operators are the principle devices of the proposed method. Although here the concept of the L^2-projection operators is demonstrated in one-dimensional problems, it is equally applicable to multidimensional problems, as shown in Section 6.3.

The L^2-projection operator is defined as $\mathcal{P}_X : L^2 \to X \subseteq H^1$ (where X is a closed (finite-dimensional approximation) space of H^1) satisfying

$$(u, v) = (\mathcal{P}_X u, v), \tag{6.8}$$

for any $v \in X$. For convenience, we denote $\mathcal{P}_X u_x$ by $\mathcal{D}_X u$, namely $\mathcal{D}_X := \mathcal{P}_X \partial_x : H^1 \to X$. We regard $\mathcal{D}_X^p (:= (\mathcal{D}_X)^p)$ $(p \geq 1)$ as the operator that approximates ∂_x^p. The following formula is straightforward.

Lemma 6.1. For any $u \in H^1$ and $v \in X$, the following holds

$$\left(\mathcal{D}_X^p u, v\right) = \left((\mathcal{D}_X^{p-1} u)_x, v\right) \qquad (p \geq 1). \tag{6.9}$$

Proof. The relation (6.9) follows from (6.8):

$$\left(\mathcal{D}_X^p u, v\right) = \left(\mathcal{P}_X (\mathcal{D}_X^{p-1} u)_x, v\right) = \left((\mathcal{D}_X^{p-1} u)_x, v\right).$$

□

The operator $\mathcal{D}_X$ can operate on any functions in H^1 any number of times (note that $\mathcal{D}_X^p : H^1 \to X$). The following equalities, obtained from (6.9), demonstrate that $\mathcal{D}_X$ is skew-symmetric and $\mathcal{D}_X^2$ is symmetric, corresponding to the skew-symmetry of ∂_x and symmetry of ∂_x^2, respectively.

Corollary 6.1. For any $u \in X$ and $v \in X$, if $[uv]_0^L = 0$, the following holds

$$(\mathcal{D}_X u, v) = -(u, \mathcal{D}_X v).$$

Also assume that $[(\mathcal{D}_X u)v]_0^L = [u(\mathcal{D}_X v)]_0^L = 0$. Then we have

$$\left(\mathcal{D}_X^2 u, v\right) = \left(u, \mathcal{D}_X^2 v\right).$$

Proof. These properties can be proved by (6.9) and integration-by-parts. □

Provided that the boundary conditions are periodic, the above operators are sufficient for the new method. However, as will be shown, multiple different boundary conditions require a more sophisticated treatment. As an extension of $\mathcal{P}_X$, we define an operator $\mathcal{P}_{X(Y)} : L^2 \to X \subseteq H^1$ (where $X(\subseteq Y)$ and Y are closed (finite-dimensional approximation) spaces of H^1) satisfying

$$\left(\mathcal{P}_{X(Y)} u, v\right) = (u, v) \tag{6.10}$$

for any $v \in Y \subseteq H^1$. Accordingly, we define an operator $\mathcal{D}_{X(Y)}$ by $\mathcal{D}_{X(Y)} := \mathcal{P}_{X(Y)} \partial_x : H^1 \to X$.

Lemma 6.2. It follows that for any $u \in H^1$ and $v \in Y$,

$$\left(\mathcal{D}_{X(Y)} u, v\right) = (u_x, v). \tag{6.11}$$

Although the operators $\mathcal{P}_{X(Y)}$ and $\mathcal{D}_{X(Y)}$ can no longer be considered "projection" operators, we can regard them as extensions of $\mathcal{P}_X$ and $\mathcal{D}_X$. Thus, we refer to $\mathcal{P}_{X(Y)}$ and $\mathcal{D}_{X(Y)}$ as "L^2-projection" operators.

6.2.2 Proposed method for Type 1 PDEs

We now apply the new method to the dissipative equation (5.13) with $H = H(u, u_x, u_{xx})$ and $s = 0$:

$$u_t = -\frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, u_x, u_{xx}) \tag{6.12}$$

as a working example, which is sufficient to show the essential idea. Note that in this case, the variational derivative $\delta H / \delta u$ becomes

$$\frac{\delta \mathcal{H}}{\delta u} := \frac{\partial H}{\partial u} - \partial_x \frac{\partial H}{\partial u_x} + \partial_x^2 \frac{\partial H}{\partial u_{xx}}.$$

Design of dissipative schemes: PHASE 1

We describe the procedure of Phase 1 (derivation of dissipative schemes) for (6.12).

Step 1′

First, we state that finding a dissipative H^1-weak form for (6.12) is not straightforward. In fact, because (6.12) contains a $\partial H/\partial u_{xx}$ term, a weak form of this equation would require H^2 elements. Instead, motivated by the construction of Weak form 1, we consider the following formulation, obtained by integrating each term by parts *only up to once*. We find u such that for any v,

$$\begin{aligned}(u_t, v) &= -\left(\frac{\partial H}{\partial u}, v\right) + \left(\partial_x \frac{\partial H}{\partial u_x}, v\right) - \left(\partial_x^2 \frac{\partial H}{\partial u_{xx}}, v\right) \\ &= -\left(\frac{\partial H}{\partial u}, v\right) - \left(\frac{\partial H}{\partial u_x}, v_x\right) + \left[\frac{\partial H}{\partial u_x} v\right]_0^L + \left(\partial_x \frac{\partial H}{\partial u_{xx}}, v_x\right) - \left[\left(\partial_x \frac{\partial H}{\partial u_{xx}}\right) v\right]_0^L.\end{aligned}$$

Note that, by virtue of the restricted integration-by-parts, the fourth term on the most right-hand side is not

$$-\left(\frac{\partial H}{\partial u_{xx}}, v_{xx}\right)$$

as in the standard finite-element formulation. This formulation remains viable in H^2 (or smoother spaces) but not in H^1. By following the rules below, we define the following *f*ormal weak form, in which the test functions alone are in H^1.

Rules for defining formal weak forms

(R1′ a) Eliminate all derivatives before the partial derivatives by introducing intermediate functions

$$q = \partial_x \frac{\partial H}{\partial u_{xx}}, \qquad r = \partial_x^2 \frac{\partial H}{\partial u_{xxx}}, \ \ldots,$$

and their associated equations, such that the test functions contain only first-order derivatives (this step should be done *r*ecursively, as required).

(R1′ b) Leave other derivatives untouched.

In our working example, applying these rules to the above formulation yields the following formal weak form. We replace $\partial_x(\partial H/\partial u_{xx})$ with q and add (6.14) by rule (R1′ a), leaving all other terms intact by (R1′ b).

Formal weak form 1. Suppose that $u(0,\cdot)$ is given. We find u, q such that for any v_1, v_2,

$$(u_t, v_1) = -\left(\frac{\partial H}{\partial u}, v_1\right) - \left(\frac{\partial H}{\partial u_x}, (v_1)_x\right) + \left[\frac{\partial H}{\partial u_x} v_1\right]_0^L + (q, (v_1)_x) - [q v_1]_0^L, \tag{6.13}$$

$$(q, v_2) = -\left(\frac{\partial H}{\partial u_{xx}}, (v_2)_x\right) + \left[\frac{\partial H}{\partial u_{xx}} v_2\right]_0^L. \tag{6.14}$$

As is obvious from the construction, the above formulation is not formulated in H^1 space, and thus is only formally valid (hence the term *formal* weak form). Note that if we ignore this defect, the dissipation property can be explicitly obtained by formal calculations. Under the assumptions:

$[qu_t]_0^L = 0$, $\left[\frac{\partial H}{\partial u_x}u_t\right]_0^L = 0$ and $\left[\frac{\partial H}{\partial u_{xx}}u_{xt}\right]_0^L = 0$,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,u_x,u_{xx})\,\mathrm{d}x =& \left(\frac{\partial H}{\partial u},u_t\right)+\left(\frac{\partial H}{\partial u_x},u_{xt}\right)+\left(\frac{\partial H}{\partial u_{xx}},u_{xxt}\right)\\ =& \left(\frac{\partial H}{\partial u},u_t\right)+\left(\frac{\partial H}{\partial u_x},u_{xt}\right)-(q,u_{xt})+\left[\frac{\partial H}{\partial u_{xx}}u_{xt}\right]_0^L\\ =& -\|u_t\|^2+\left[\frac{\partial H}{\partial u_x}u_t\right]_0^L-[qu_t]_0^L\le 0. \end{aligned} \tag{6.15}$$

Here the first equality is obtained by the chain rule. The second follows from (6.14) with $v_2 = u_{xt}$, and the third from (6.13) with $v_1 = u_t$.

Remark 6.2. Other definitions of formal weak forms for which the dissipation property can be formally checked are possible. However, the definition of formal weak forms following the proposed rule is ideally suitable for implementing Step 2′ (presented below).

Step 2′

In this step, the above formal weak form is spatially discretised such that the semi-discrete scheme is consistent in some finite-dimensional approximation spaces of H^1. The rules are described below.

Rules for constructing semi-discrete schemes

(R2′ a) Set finite-dimensional trial and test function spaces for solutions of the formal weak form;

(R2′ b) Replace derivatives in H and in partial derivatives with $\mathcal{D}_{S_j(W_j)}$ by introducing function spaces S_j's, W_j's as necessary such that

$$u_x \to \mathcal{D}_{S_j(W_j)}u, \qquad u_{xx} \to \mathcal{D}_{S_{j+1}(W_{j+1})}\mathcal{D}_{S_j(W_j)}u, \ \dots$$

(i.e., introduce new function spaces for each additional derivative);

(R2′ c) Place projection operators before the partial derivatives by introducing new function spaces for each partial derivative;

(R2′ d) Leave remaining derivatives (mainly in test functions) intact.

Hereafter, where their meanings are unambiguous, we denote $\mathcal{P}_{S_j(W_j)}$ and $\mathcal{D}_{S_j(W_j)}$ by the simpler forms $\mathcal{P}_j$ and $\mathcal{D}_j$, respectively.

Semi-discrete scheme 2 (with L^2-projection operators)**.** Suppose that $u(0,\cdot)$ is given in S_1. We find $u(t,\cdot)\in S_1, q\in S_2$ such that for any $v_1\in W_1, v_2\in W_2$,

$$(u_t,v_1) = -\left(\frac{\partial H}{\partial u},v_1\right)-\left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3u)},(v_1)_x\right)+\left[\left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3u)}\right)v_1\right]_0^L+(q,(v_1)_x)-[qv_1]_0^L, \tag{6.16}$$

$$(q,v_2) = -\left(\mathcal{P}_6\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3u)},(v_2)_x\right)+\left[\left(\mathcal{P}_6\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3u)}\right)v_2\right]_0^L, \tag{6.17}$$

where $H = H(u,\mathcal{D}_3u,\mathcal{D}_4\mathcal{D}_3u)$.

Following (R2′ a), we introduce trial function spaces S_1 and S_2 for $u(t,\cdot)$ and q, respectively, and corresponding test function spaces W_1 and W_2. Following (R2′ b), we replace derivatives with S_3, W_3, S_4, and W_4 such that

$$u_x \to \mathcal{D}_3u, \qquad u_{xx} \to \mathcal{D}_4\mathcal{D}_3u.$$

Following (R2′ c), we introduce trial and test functions S_5, W_5, S_6, and W_6 for the projection operators $\mathcal{P}_5$ and $\mathcal{P}_6$, which is placed before the partial derivatives. Provided that the rules are obeyed, the numbering is arbitrary. The following proposition specifies sufficient conditions for the dissipation property in each given numbering.

Remark 6.3. In the proposed method, the semi-discrete scheme is consistent in H^1 by virtue of (R2′ a), (R2′ b), and (R2′ d). Rule (R2′ c) is necessary for the dissipation property.

Remark 6.4. The proposed method imposes many (test and trial) function spaces. Each space must be selected to comply with the boundary conditions and satisfy the assumptions of the following proposition. Selection can be based on the standard theory of finite element methods. An example is given in Section 6.2.4.

Remark 6.5. Some remarks on the notation: $\partial H/\partial(\mathcal{D}_j u)$ denotes the substitution of $u_x, u_{xx}, \dots$ in $\partial H/\partial u_x$ by $\mathcal{D}_j u, \mathcal{D}_{j+1}\mathcal{D}_j u, \dots$, and similarly for $\partial H/\partial u_{xx}, \partial H/\partial u_{xxx}, \dots$. For example, given a function $H(u, u_x, u_{xx}) = u u_x u_{xx}$,

$$\frac{\partial H}{\partial u} = (\mathcal{D}_j u)(\mathcal{D}_{j+1}\mathcal{D}_j u), \qquad \frac{\partial H}{\partial(\mathcal{D}_j u)} = u(\mathcal{D}_{j+1}\mathcal{D}_j u), \qquad \frac{\partial H}{\partial(\mathcal{D}_{j+1}\mathcal{D}_j u)} = u(\mathcal{D}_j u).$$

Proposition 6.3 (Semi-discrete scheme 2: dissipation property)**.** Assume that the boundary conditions satisfy

$$\left[\left(\mathcal{P}_6 \frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3 u)}\right)\mathcal{D}_3 u_t\right]_0^L = 0, \qquad \left[\left(\mathcal{P}_5 \frac{\partial H}{\partial(\mathcal{D}_3 u)}\right)u_t\right]_0^L = 0, \qquad [q u_t]_0^L = 0.$$

Also assume that $S_5 \subseteq W_3$, $S_6 \subseteq W_4$, $S_2 \subseteq W_3$, $\mathcal{D}_3 u_t \in W_5$, $\mathcal{D}_4\mathcal{D}_3 u_t \in W_6$, $\mathcal{D}_3 u_t \in W_2$,$u_t \in W_1$ and $u_x, \mathcal{D}_3 u \in C^1(\mathbb{R}^+; L^2(0, L))$. Then the solution of Semi-discrete scheme 2 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u, \mathcal{D}_3 u, \mathcal{D}_4\mathcal{D}_3 u)\,\mathrm{d}x \le 0.$$

Proof. The proof is similar to the formal calculation of (6.15) and involves carefully checking each equality.

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u, \mathcal{D}_3 u, \mathcal{D}_4\mathcal{D}_3 u)\,\mathrm{d}x \\
&= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial(\mathcal{D}_3 u)}, \mathcal{D}_3 u_t\right) + \left(\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3 u)}, \mathcal{D}_4\mathcal{D}_3 u_t\right) \\
&= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3 u)}, \mathcal{D}_3 u_t\right) + \left(\mathcal{P}_6\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3 u)}, \mathcal{D}_4\mathcal{D}_3 u_t\right) \\
&= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3 u)}, u_{xt}\right) + \left(\mathcal{P}_6\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3 u)}, (\mathcal{D}_3 u_t)_x\right) \\
&= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3 u)}, u_{xt}\right) - (q, \mathcal{D}_3 u_t) + \left[\left(\mathcal{P}_6\frac{\partial H}{\partial(\mathcal{D}_4\mathcal{D}_3 u)}\right)\mathcal{D}_3 u_t\right]_0^L \\
&= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3 u)}, u_{xt}\right) - (q, u_{xt}) = -\|u_t\|^2 + \left[\left(\mathcal{P}_5\frac{\partial H}{\partial(\mathcal{D}_3 u)}\right)u_t\right]_0^L - [q u_t]_0^L \le 0.
\end{aligned}$$

The first equality is obtained by the chain rule. Eq. (6.10) is used in the second and third terms of the second equality, as permitted by the assumption $\mathcal{D}_3 u_t \in W_5$ for the second term and $\mathcal{D}_4\mathcal{D}_3 u_t \in W_6$ for

the third term. Note that the terms included in the partial derivatives remain in L^2: by the Sobolev embedding theorem, $S_3,\ S_4 \subset H^1 \subset C^0$ (for example, [7, Theorem 7.3.8]).

The third equality follows from (6.11), which is allowed by the assumptions $S_5 \subseteq W_3$ and $S_6 \subseteq W_4$. The fourth follows from (6.16) with $v_2 = \mathcal{D}_3 u_t \in W_2$. The fifth equality again uses (6.11) as allowed by the assumption $S_2 \subseteq W_3$, while the sixth equality derives from (6.17) with $v_1 = u_t \in W_1$. □

Remark 6.6. The procedure of this step was designed for completely automatic implementation, but can be slightly modified if necessary. Above, we considered an energy term of the form

$$\int_0^L H(u, \mathcal{D}_3 u, \mathcal{D}_4\mathcal{D}_3 u)\,\mathrm{d}x.$$

However, other definitions may be possible. For example, if the energy takes the form

$$\int_0^L H(u, u_x, \mathcal{D}_4\mathcal{D}_3 u)\,\mathrm{d}x,$$

an intended semi-discrete scheme can also be derived.

Step 3

In this step, we temporally discretise Semi-discrete scheme 2, such that the dissipation property is retained. We adopt the following notation for convenience. Hereafter we also call the discrete quantities

$$\frac{\partial H_\mathrm{d}}{\partial(u^{(n+1)}, u^{(n)})}, \qquad \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_j u^{(n+1)}, \mathcal{D}_j u^{(n)})}, \qquad \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_{j+1}\mathcal{D}_j u^{(n+1)}, \mathcal{D}_{j+1}\mathcal{D}_j u^{(n)})},$$

corresponding to $\partial H/\partial u$, $\partial H/\partial u_x$ and $\partial H/\partial u_{xx}$, respectively, the discrete partial derivatives, provided that they satisfy the following discrete chain rule:

$$\begin{aligned}
&\frac{1}{\Delta t}\int_0^L \left(H(u^{(n+1)}, \mathcal{D}_j u^{(n+1)}, \mathcal{D}_{j+1}\mathcal{D}_j u^{(n+1)}) - H(u^{(n)}, \mathcal{D}_j u^{(n)}, \mathcal{D}_{j+1}\mathcal{D}_j u^{(n)})\right)\mathrm{d}x \\
&= \left(\frac{\partial H_\mathrm{d}}{\partial(u^{(n+1)}, u^{(n)})}, \frac{u^{(n+1)} - u^{(n)}}{\Delta t}\right) + \left(\frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_j u^{(n+1)}, \mathcal{D}_j u^{(n)})}, \frac{\mathcal{D}_j u^{(n+1)} - \mathcal{D}_j u^{(n)}}{\Delta t}\right) \\
&\quad + \left(\frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_{j+1}\mathcal{D}_j u^{(n+1)}, \mathcal{D}_{j+1}\mathcal{D}_j u^{(n)})}, \frac{\mathcal{D}_{j+1}\mathcal{D}_j u^{(n+1)} - \mathcal{D}_{j+1}\mathcal{D}_j u^{(n)}}{\Delta t}\right),
\end{aligned}$$

cf. Definition 6.1 for the standard case. Using these discrete partial derivatives, we define a dissipative scheme as follows.

Scheme 2 (dissipative H^1-Galerkin schemes for (6.12))**.** Suppose that $u^{(0)}$ is given in S_1. We find $u^{(n+1)} \in S_1$ and $q^{(n+\frac{1}{2})} \in S_2$ $(n = 0, 1, \dots)$ such that for any $v_1 \in W_1$ and $v_2 \in W_2$,

$$\begin{aligned}
\left(\frac{u^{(n+1)} - u^{(n)}}{\Delta t}, v_1\right) &= -\left(\frac{\partial H_\mathrm{d}}{\partial(u^{(n+1)}, u^{(n)})}, v_1\right) - \left(\mathcal{P}_5 \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_3 u^{(n+1)}, \mathcal{D}_3 u^{(n)})}, (v_1)_x\right) \\
&\quad + \left[\mathcal{P}_5 \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_3 u^{(n+1)}, \mathcal{D}_3 u^{(n)})} v_1\right]_0^L + \left(q^{(n+\frac{1}{2})}, (v_1)_x\right) - \left[q^{(n+\frac{1}{2})} v_1\right]_0^L, \\
\left(q^{(n+\frac{1}{2})}, v_2\right) &= -\left(\mathcal{P}_6 \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_4\mathcal{D}_3 u^{(n+1)}, \mathcal{D}_4\mathcal{D}_3 u^{(n)})}, (v_2)_x\right) + \left[\left(\mathcal{P}_6 \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_4\mathcal{D}_3 u^{(n+1)}, \mathcal{D}_4\mathcal{D}_3 u^{(n)})}\right) v_2\right]_0^L.
\end{aligned}$$

The following theorem, which states that the scheme is dissipative, immediately follows from the scheme construction.

Theorem 6.2 (Scheme 2: dissipation property)**.** Assume that the boundary conditions satisfy

$$\left[\left(\mathcal{P}_6\frac{\partial H_{\mathrm{d}}}{\partial(\mathcal{D}_4\mathcal{D}_3u^{(n+1)},\mathcal{D}_4\mathcal{D}_3u^{(n)})}\right)\left(\frac{\mathcal{D}_3u^{(n+1)}-\mathcal{D}_3u^{(n)}}{\Delta t}\right)\right]_0^L=0,$$
$$\left[\left(\mathcal{P}_5\frac{\partial H_{\mathrm{d}}}{\partial(\mathcal{D}_3u^{(n+1)},\mathcal{D}_3u^{(n)})}\right)\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right)\right]_0^L=0,\qquad \left[q^{(n+\frac{1}{2})}\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}\right)\right]_0^L=0.$$

Also assume that $S_5\subseteq W_3$, $S_6\subseteq W_4$, $S_2\subseteq W_3$, $(\mathcal{D}_3u^{(n+1)}-\mathcal{D}_3u^{(n)})/\Delta t\in W_5$, $(\mathcal{D}_4\mathcal{D}_3u^{(n+1)}-\mathcal{D}_4\mathcal{D}_3u^{(n)})/\Delta t\in W_6$, $(\mathcal{D}_3u^{(n+1)}-\mathcal{D}_3u^{(n)})/\Delta t\in W_2$ and $(u^{(n+1)}-u^{(n)})/\Delta t\in W_1$. Then the solution of Scheme 2 satisfies

$$\frac{1}{\Delta t}\int_0^L\left(H(u^{(n+1)},\mathcal{D}_3u^{(n+1)},\mathcal{D}_4\mathcal{D}_3u^{(n+1)})-H(u^{(n)},\mathcal{D}_3u^{(n)},\mathcal{D}_4\mathcal{D}_3u^{(n)})\right)\mathrm{d}x\le 0,\qquad n=0,1,2,\dots.$$

Design of dissipative schemes: PHASE 2

In Phase 1 (Steps 1′-3), we designed a dissipative H^1-Galerkin scheme. In the remaining steps, we confirm the validity of the scheme by finding an underlying H^1-weak form of Scheme 2.

Step 4

In this step, we express Semi-discrete scheme 2 in a more familiar form, eliminating the L^2-projection operators. No new concepts are introduced at this stage; the semi-discrete scheme is merely rewritten. Hence, the following Semi-discrete scheme 2′ is mathematically equivalent to Semi-discrete scheme 2. This step is implemented on the basis of the definition of the L^2-projection operators. For instance, introducing an intermediate function $a_1\in S_3$, we rewrite the term $\partial H/\partial(\mathcal{D}_3u)$ as $\partial H/\partial a_1$, as adding a new equation $(a_1,v)=(u_x,v)$ for any $v\in W_3$ (see Lemma 6.2).

Semi-discrete scheme 2′ (without L^2-projection operators)**.** Suppose that $u(0,\cdot)$ is given in S_1. We find $u(t,\cdot)\in S_1$, $q\in S_2$, $a_1\in S_3$, $a_2\in S_4$, $r_1\in S_5$, $r_2\in S_6$ such that for any $v_1\in W_1$, $v_2\in W_2$, $v_3\in W_3$, $v_4\in W_4$, $v_5\in W_5$, $v_6\in W_6$,

$$\begin{aligned}
&(u_t,v_1)=-\left(\frac{\partial H}{\partial u},v_1\right)-(r_1,(v_1)_x)+[r_1v_1]_0^L+(q,(v_1)_x)-[qv_1]_0^L,\\
&(q,v_2)=-(r_2,(v_2)_x)+[r_2v_2]_0^L,\\
&(a_1,v_3)=(u_x,v_3),\\
&\big(a_2,v_4\big)=\big((a_1)_x,v_4\big),\\
&(r_1,v_5)=\left(\frac{\partial H}{\partial a_1},v_5\right),\\
&(r_2,v_6)=\left(\frac{\partial H}{\partial a_2},v_6\right).
\end{aligned}$$

Proposition 6.4 (Semi-discrete scheme 2′: dissipation property)**.** Assume that the boundary conditions satisfy

$$[r_2(a_1)_t]_0^L=0,\qquad [r_1u_t]_0^L=0,\qquad [qu_t]_0^L=0.$$

Also assume that $S_5\subseteq W_3$, $S_6\subseteq W_4$, $S_2\subseteq W_3$, $(a_1)_t\in W_5$, $(a_2)_t\in W_6$, $(a_1)_t\in W_2$, $u_t\in W_1$ and $u_x,(a_1)_x\in C^1(\mathbb{R}^+;L^2(0,L))$. Then the solution of Semi-discrete scheme 2′ satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,a_1,a_2)\,\mathrm{d}x\le 0.$$

Proof. The proof is omitted since we have proved Proposition 6.3. □

Step 5

In this step, all finite-dimensional function spaces S_i's and W_i's are replaced by their corresponding infinite-dimensional function spaces S_i^c's and W_i^c's, which are subspaces of $H^1(0,L)$. This replacement, which is valid for commonly used finite-dimensional subspaces, yields the following weak form.

Weak form 2. We find $u(t,\cdot)\in S_1^c$, $q\in S_2^c$, $a_1\in S_3^c$, $a_2\in S_4^c$, $r_1\in S_5^c$, $r_2\in S_6^c$ such that for any $v_1\in W_1^c$, $v_2\in W_2^c$, $v_3\in W_3^c$, $v_4\in W_4^c$, $v_5\in W_5^c$, $v_6\in W_6^c$,

$$
\begin{aligned}
&(u_t,v_1)=-\left(\frac{\partial H}{\partial u},v_1\right)-(r_1,(v_1)_x)+[r_1v_1]_0^L+(q,(v_1)_x)-[qv_1]_0^L,\\
&(q,v_2)=-(r_2,(v_2)_x)+[r_2v_2]_0^L,\\
&(a_1,v_3)=(u_x,v_3),\\
&\big(a_2,v_4\big)=\big((a_1)_x,v_4\big),\\
&(r_1,v_5)=\left(\frac{\partial H}{\partial a_1},v_5\right),\\
&(r_2,v_6)=\left(\frac{\partial H}{\partial a_2},v_6\right).
\end{aligned}
$$

Clearly, Weak form 2 is consistent in H^1, and has the dissipation property.

Proposition 6.5 (Weak form 2: dissipation property)**.** Assume that the boundary conditions satisfy

$$[r_2(a_1)_t]_0^L=0,\qquad [r_1u_t]_0^L=0,\qquad [qu_t]_0^L=0.$$

Also assume that $S_5^c\subseteq W_3^c$, $S_6^c\subseteq W_4^c$, $S_2^c\subseteq W_3^c$, $(a_1)_t\in W_5^c$, $(a_2)_t\in W_6^c$, $(a_1)_t\in W_2^c$, $u_t\in W_1^c$ and $u_x,(a_1)_x\in C^1(\mathbb{R}^+;L^2(0,L))$. Then the solution of Weak form 2 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,a_1,a_2)\,\mathrm{d}x\le 0.$$

Thus, we have found the desired underlying weak form, which is consistent in H^1 and retains the dissipation property. Clearly, this form and the five intermediate functions from which it is derived are not easily determined from the original PDE (6.12). Simply by following the proposed approach, we have automatically extracted the desired dissipative scheme and the underlying weak form, which highlights the power of the new method.

Step 6

Finally, we must check the relationship between (6.12) and Weak form 2. In fact, Weak form 2 is the natural weak formulation of the system of equations

$$
\begin{aligned}
&u_t=-\frac{\partial H}{\partial u}+(r_1)_x-q_x,\qquad && q=(r_2)_x,\qquad && a_1=u_x,\\
&a_2=(a_1)_x, && r_1=\frac{\partial H}{\partial a_1},\qquad && r_2=\frac{\partial H}{\partial a_2},
\end{aligned}
$$

which is equivalent to (6.12).

6.2.3 Proposed method for Type 2 PDEs

We now apply the new method to the conservative equation:

$$u_t = \mathcal{S}\frac{\delta \mathcal{H}}{\delta u}, \tag{6.18}$$

where $\mathcal{S} = \mathcal{S}(u, u_x, u_{xx}, \dots, \partial_x, \partial_x^2, \dots)$ is skew-symmetric and polynomial with respect to $u, u_x, u_{xx}, \dots, \partial_x, \partial_x^2, \dots$. For simplicity, we assume that $H = H(u, u_x)$.

To clarify the essential idea of the proposed method, we restrict our attention to the conservative case and assume periodic boundary conditions (see Remark 6.8). Moreover, we will describe the framework with the toy problem $\mathcal{S} = (u_{xx}\partial_x + \partial_x u_{xx}) + \partial_x^3$ (which, as in the usual interpretation, operates on a function f such that $(g\partial_x + \partial_x g)f = gf_x + \partial_x(gf)$). This operator includes two typical forms of complicated differential operators:

- $\mathcal{S}$ contains not only ∂_x^s but also functions of $u, u_x, \dots$,
- $\mathcal{S}$ is the summation of differential operators.

Current research frequently involves operators that are considerably more complex than these, such as inverses of differential operators. We will demonstrate the treatment of such complex operators with a concrete example in the next subsection.

Design of conservative schemes: PHASE 1

In this subsection, we apply the Phase 1 procedure (derivation of dissipative schemes) to (6.18).

Step 1′

Consider the following formulation. Since this formulation may not valid in H^1 (depending on the operator $\mathcal{S}$), we call it the formal weak form.

Formal weak form 2. Suppose that $u(0, \cdot)$ is given. We find u, p such that for any v_1, v_2,

$$\begin{aligned}(u_t, v_1) &= (\mathcal{S}p, v_1),\\ (p, v_2) &= \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right).\end{aligned}$$

If the operator $\mathcal{S}$ is skew-symmetric, the conservation property can be obtained by formal calculation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u, u_x)\,\mathrm{d}x = \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) = (p, u_t) = (\mathcal{S}p, p) = 0. \tag{6.19}$$

Step 2′

In this step, the above formal weak form is spatially discretised such that the semi-discrete scheme is consistent within an approximation space X_{p} of $H^1(\mathbb{T})$. This step is completely automatic and achieved by replacing differential operators with $\mathcal{D}_{X_{\mathrm{p}}}$; i.e., replacing $\mathcal{S}(u_{xx}, \partial_x, \partial_x^3)$ with $\mathcal{S}_{\mathrm{sd}} = \mathcal{S}(\mathcal{D}^2_{X_{\mathrm{p}}}u, \mathcal{D}_{X_{\mathrm{p}}}, \mathcal{D}^3_{X_{\mathrm{p}}})$. Note that for the above-defined $\mathcal{S}(u_{xx}, \partial_x, \partial_x^3)$, $\mathcal{S}(\mathcal{D}^2_{X_{\mathrm{p}}}u, \mathcal{D}_{X_{\mathrm{p}}}, \mathcal{D}^3_{X_{\mathrm{p}}})$ is skew-symmetric.

Semi-discrete scheme 3 (with L^2-projection operators)**.** Suppose that $u(0, \cdot)$ is given in X_{p}. We find $u(t, \cdot) \in X_{\mathrm{p}}$ and $p \in X_{\mathrm{p}}$ such that for any $v_1 \in X_{\mathrm{p}}$ and $v_2 \in X_{\mathrm{p}}$,

$$\begin{aligned}(u_t, v_1) &= (\mathcal{S}_{\mathrm{sd}}p, v_1),\\ (p, v_2) &= \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right)\end{aligned}$$

where $H = H(u, u_x)$.

Note that replacing the operator $\mathcal{S}$ with $\mathcal{S}_{\mathrm{sd}}$ renders the scheme consistent in H^1 space.

Proposition 6.6 (Semi-discrete scheme 3: conservation property)**.** The solution of Semi-discrete scheme 3 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u,u_x)\,\mathrm{d}x = 0.$$

Proof. Since $\mathcal{S}_{\mathrm{sd}}$ is valid in H^1 space, the calculation (6.19) becomes mathematically rigorous rather than merely formal. □

Step 3

In this step, we temporally discretise the obtained semi-discrete scheme. Since this step merely discretise the gradients by the discrete gradient method, we omit the details and show only the result.

Scheme 3 (Conservative H^1-Galerkin schemes for (6.18))**.** Suppose that $u^{(0)}$ is given in X_{p}. We find $u^{(n+1)} \in X_{\mathrm{p}}$ and $q^{(n+\frac{1}{2})} \in X_{\mathrm{p}}$ $(n = 0,1,\dots)$ such that for any $v_1 \in X_{\mathrm{p}}$ and $v_2 \in X_{\mathrm{p}}$,

$$\begin{aligned}
&\left(\frac{u^{(n+1)}-u^{(n)}}{\Delta t}, v_1\right) = \left(\mathcal{S}_{\mathrm{d}} p^{(n+\frac{1}{2})}, v_1\right),\\
&\left(p^{(n+\frac{1}{2})}, v_2\right) = \left(\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)},u^{(n)})}, v_2\right) + \left(\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)},u_x^{(n)})}, (v_2)_x\right),
\end{aligned}$$

where $\mathcal{S}_{\mathrm{d}} = \mathcal{S}(\mathcal{D}^2_{X_{\mathrm{p}}} u^{(n+\frac{1}{2})}, \mathcal{D}_{X_{\mathrm{p}}}, \mathcal{D}^3_{X_{\mathrm{p}}})$ and $u^{(n+\frac{1}{2})} = (u^{(n+1)} + u^{(n)})/2$.

Theorem 6.3 (Scheme 3: conservation property)**.** The solution of Scheme 3 satisfies

$$\frac{1}{\Delta t}\int_{\mathbb{T}} \left(H(u^{(n+1)}, u_x^{(n+1)}) - H(u^{(n)}, u_x^{(n)})\right)\mathrm{d}x = 0, \qquad n = 0,1,2,\dots.$$

Design of conservative schemes: PHASE 2

Step 4

Recall that $\mathcal{S}_{\mathrm{sd}} p = \mathcal{S}(\mathcal{D}^2_{X_{\mathrm{p}}} u, \mathcal{D}_{X_{\mathrm{p}}}, \mathcal{D}^3_{X_{\mathrm{p}}})p = (\mathcal{D}^2_{X_{\mathrm{p}}} u)(\mathcal{D}_{X_{\mathrm{p}}} p) + \mathcal{D}_{X_{\mathrm{p}}}((\mathcal{D}^2_{X_{\mathrm{p}}} u)p) + \mathcal{D}^3_{X_{\mathrm{p}}} p$. Introducing new variables

$$\begin{aligned}
&u_1 := \mathcal{D}_{X_{\mathrm{p}}} u, \qquad u_2 := \mathcal{D}_{X_{\mathrm{p}}} u_1,\\
&p_1 := \mathcal{D}_{X_{\mathrm{p}}} p, \qquad p_2 := \mathcal{D}_{X_{\mathrm{p}}} p_1, \qquad p_3 := \mathcal{D}_{X_{\mathrm{p}}} p_2,\\
&q := \mathcal{D}_{X_{\mathrm{p}}}(u_2 p),
\end{aligned}$$

we can rewrite Semi-discrete scheme 3 in a more familiar form as follows.

Semi-discrete scheme 3′ (without L^2-projection operators)**.** Suppose that $u(0,\cdot)$ is given in X_{p}. We find $u(t,\cdot), u_1, u_2, p, p_1, p_2, p_3, q \in X_{\mathrm{p}}$ such that for any $v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8 \in X_{\mathrm{p}}$,

$$\begin{aligned}
&(u_t, v_1) = (u_2 p_1 + q + p_3, v_1),\\
&(p, v_2) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right),\\
&(u_1, v_3) = (u_x, v_3), \qquad (u_2, v_4) = ((u_1)_x, v_4),\\
&(p_1, v_5) = (p_x, v_5), \qquad (p_2, v_6) = ((p_1)_x, v_6), \qquad (p_3, v_7) = ((p_2)_x, v_7),\\
&(q, v_8) = ((u_2 p)_x, v_8).
\end{aligned}$$

Steps 5 and 6

A conservative H^1-weak form is easily obtained by converting X_p in $3'$ into $H^1(\mathbb{T})$, and confirming its conformity to the original equation (6.18). The calculations are straightforward but tedious, and are hence omitted. As commented at the end of Section 6.2.2, we emphasize that the weak form and its seven intermediate functions are not at all obvious from the original PDE.

Remark 6.7. In illustrating the proposed method, we defined $\mathcal{S}_\mathrm{sd} = \mathcal{S}(\mathcal{D}^2_{X_\mathrm{p}} u, \mathcal{D}_{X_\mathrm{p}}, \mathcal{D}^3_{X_\mathrm{p}})$. Other definitions such as $\mathcal{S}(\mathcal{D}^2_{X_\mathrm{p}} u, \partial_x, \mathcal{D}^3_{X_\mathrm{p}})$ can be adopted, provided that they are consistent with H^1 and are skew-symmetric.

Remark 6.8. For PDEs containing higher-order derivatives in H and complicated $\mathcal{S}$, the desired Galerkin schemes can be derived by combining the proposed methods developed for Types 1 and 2. Similarly, general boundary conditions can be treated as discussed in the procedure for the Type 1 PDEs.

6.2.4 Applications of the proposed method

This subsection is devoted to the applications of the proposed method. Firstly, we treat the Swift–Hohenberg equation, a dissipative equation of Type 1, and discusses its boundary conditions. Next, we consider the Kawahara equation, a conservative equation of Type 1, discuss the implementation issues, and present the result of numerical experiments. Finally, we focus on the Camassa–Holm equation, a conservative PDE of Type 2.

Type 1: The Swift–Hohenberg equation

The Swift–Hohenberg equation is a form of (6.12) with $H(u, u_x, u_{xx}) = -u^2 + u^4/4 - u_x^2 + u_{xx}^2/2$, which usually assumes the following boundary conditions

$$u_x = u_{xxx} = 0 \qquad \text{at } x = 0, L, \tag{6.20}$$

or

$$u_t = u_{xx} = 0 \qquad \text{at } x = 0, L. \tag{6.21}$$

In either case, the classical solution is easily shown to be energy-dissipative. Applying the procedure developed in Section 6.2.2, we automatically obtain the formal weak form, semi-discrete scheme and fully discrete scheme. Here we demonstrate the selection of the function spaces (based on standard finite element theory, as mentioned in Remark 6.4). Let $S_h \subset H^1(0, L)$ be a piecewise linear function space over the grids. Let $S_{h,0} = \{v \mid v \in S_h, v(0) = v(L) = 0\}$ and $S_{h,g} = \{v \mid v \in S_h, v(0) = v(L) = g\}$, where g is a constant. Obviously, $S_{h,0}$ corresponds to $H^1_0 = \{v \mid v \in H^1, v(0) = v(L) = 0\}$ and $S_{h,g}$ corresponds to $H^1_g = \{v \mid v \in H^1, v(0) = v(L) = g\}$. For boundary conditions (6.20), natural choices are $S_1 = W_1 = S_4 = W_4 = S_6 = W_6 = S_{h,0}$ and $S_2 = W_2 = S_3 = W_3 = S_5 = W_5 = S_h$. Correspondingly, $S_1^\mathrm{c} = W_1^\mathrm{c} = S_4^\mathrm{c} = W_4^\mathrm{c} = S_6^\mathrm{c} = W_6^\mathrm{c} = H^1_0$ and $S_2^\mathrm{c} = W_2^\mathrm{c} = S_3^\mathrm{c} = W_3^\mathrm{c} = S_5^\mathrm{c} = W_5^\mathrm{c} = H^1$. For boundary conditions (6.21), natural choices are $S_2 = W_2 = S_3 = W_3 = S_6 = W_6 = S_{h,0}$, $S_1 = W_1 = S_{h,g}$ and $S_4 = W_4 = S_5 = W_5 = S_h$, with corresponding spaces $S_2^\mathrm{c} = W_2^\mathrm{c} = S_3^\mathrm{c} = W_3^\mathrm{c} = S_6^\mathrm{c} = W_6^\mathrm{c} = H^1_0$, $S_1^\mathrm{c} = W_1^\mathrm{c} = H^1_g$ and $S_4^\mathrm{c} = W_4^\mathrm{c} = S_5^\mathrm{c} = W_5^\mathrm{c} = H^1$. These relationships are consistent with the assumptions of Proposition 6.3 and Proposition 6.5.

Type 1: The Kawahara equation

Consider a PDE of the form

$$u_t = \partial_x \frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, u_x, u_{xx}).$$

In applying the proposed method to this form of PDE, we adopt the Kawahara equation (5.15), assuming the periodic boundary conditions for simplicity. We set $S_1 = S_2 = \cdots = W_1 = W_2 = \cdots =: X_\mathrm{p} \subset H^1(\mathbb{T})$ (where $\mathbb{T}$ denotes a torus of length L). To conserve space, we omit all steps (Steps 1-6), and present only the formal weak form, the resulting Galerkin scheme, and its underlying weak form.

First, we define a formal weak form. Introducing an intermediate function p, the equation is converted to the system

$$u_t = (p_1)_x, \qquad p_1 = \frac{\delta H}{\delta u}.$$

Let us consider the following formulation, obtained by integrating-by-part up to once each term. We find u, p_1 such that for any v_1, v_2,

$$\begin{aligned}
&(u_t, v_1) = ((p_1)_x, v_1),\\
&(p_1, v_2) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right) - \left(\partial_x \frac{\partial H}{\partial u_{xx}}, (v_2)_x\right).
\end{aligned}$$

To derive the formal weak form, we follow the procedure of Section 6.2.2. We find u, p and q such that for any v_1, v_2 and v_3,

$$\begin{aligned}
&(u_t, v_1) = ((p_1)_x, v_1),\\
&(p_1, v_2) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right) - (q, (v_2)_x),\\
&(q, v_3) = -\left(\frac{\partial H}{\partial u_{xx}}, (v_3)_x\right).
\end{aligned}$$

Note that the new intermediate function q is introduced by rule (R1′ a). The conservation property can be obtained by formal calculation:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u, u_x, u_{xx})\,\mathrm{d}x =& \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) + \left(\frac{\partial H}{\partial u_{xx}}, u_{xxt}\right)\\
=& \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) - (q, u_{xt})\\
=& (p_1, u_t) = ((p_1)_x, p_1) = 0.
\end{aligned}$$

We next derive a semi-discrete scheme using the L^2-projection operators. Suppose that $u(0,\cdot)$ is given in X_p. We find $u(t,\cdot), p_1, q \in X_\mathrm{p}$ such that for any $v_1, v_2, v_3 \in X_\mathrm{p}$,

$$\begin{aligned}
&(u_t, v_1) = ((p_1)_x, v_1),\\
&(p_1, v_2) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H}{\partial (\mathcal{D}_{X_\mathrm{p}} u)}, (v_2)_x\right) - (q, (v_2)_x),\\
&(q, v_3) = -\left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H}{\partial (\mathcal{D}^2_{X_\mathrm{p}} u)}, (v_3)_x\right),
\end{aligned}$$

where

$$\frac{\partial H}{\partial u} = -\frac{u^2}{3}, \qquad \frac{\partial H}{\partial (\mathcal{D}_{X_\mathrm{p}} u)} = \alpha \mathcal{D}_{X_\mathrm{p}} u, \qquad \frac{\partial H}{\partial (\mathcal{D}^2_{X_\mathrm{p}} u)} = \beta \mathcal{D}^2_{X_\mathrm{p}} u.$$

Here following rule (R2′ b), u_x and u_{xx} in the partial derivatives are replaced by $\mathcal{D}_{X_\mathrm{p}} u$ and $\mathcal{D}^2_{X_\mathrm{p}} u$, and $\mathcal{P}_{X_\mathrm{p}}$'s are placed in front of the partial derivatives by (R2′ c). (R2′ a) is implicitly obeyed. We do not

discuss the function spaces since they are X_p on account of the periodic boundary conditions. This scheme is consistent in $X_\mathrm{p} \subset H^1(\mathbb{T})$ and has the following rigorous conservation property:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u, \mathcal{D}_{X_\mathrm{p}} u, \mathcal{D}^2_{X_\mathrm{p}} u)\,\mathrm{d}x = 0.$$

Third, we temporally discretise the above semi-discrete scheme to obtain the following fully discrete scheme. Suppose that $u^{(0)}$ is given in X_p. We find $u^{(n+1)}, p_1^{(n+\frac{1}{2})}, q^{(n+\frac{1}{2})} \in X_\mathrm{p}$ $(n = 0, 1, \dots)$ such that for any $v_1, v_2, v_3 \in X_\mathrm{p}$,

$$\left(\frac{u^{(n+1)} - u^{(n)}}{\Delta t}, v_1\right) = \left((p_1^{(n+\frac{1}{2})})_x, v_1\right), \tag{6.22}$$

$$\left(p_1^{(n+\frac{1}{2})}, v_2\right) = \left(\frac{\partial H_\mathrm{d}}{\partial(u^{(n+1)}, u^{(n)})}, v_2\right) + \left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}_{X_\mathrm{p}} u^{(n)})}, (v_2)_x\right) - \left(q^{(n+\frac{1}{2})}, (v_2)_x\right), \tag{6.23}$$

$$\left(q^{(n+\frac{1}{2})}, v_3\right) = -\left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}^2_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}^2_{X_\mathrm{p}} u^{(n)})}, (v_3)_x\right), \tag{6.24}$$

where

$$\frac{\partial H_\mathrm{d}}{\partial(u^{(n+1)}, u^{(n)})} = -\frac{(u^{(n+1)})^2 + u^{(n+1)}u^{(n)} + (u^{(n)})^2}{6},$$

$$\frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}_{X_p} u^{(n)})} = \alpha\left(\frac{\mathcal{D}_{X_\mathrm{p}} u^{(n+1)} + \mathcal{D}_{X_\mathrm{p}} u^{(n)}}{2}\right),$$

$$\frac{\partial H_\mathrm{d}}{\partial(\mathcal{D}^2_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}^2_{X_\mathrm{p}} u^{(n)})} = \beta\left(\frac{\mathcal{D}^2_{X_\mathrm{p}} u^{(n+1)} + \mathcal{D}^2_{X_\mathrm{p}} u^{(n)}}{2}\right).$$

These discrete partial derivatives are the discretised form of $\partial H/\partial u = -u^2/2$, $\partial H/\partial u_x = \alpha u_x$ and $\partial H/\partial u_{xx} = \beta u_{xx}$ (recall that $H(u, u_x, u_{xx}) = -u^3/6 + \alpha u_x^2/2 + \beta u_{xx}^2/2$) and satisfy the discrete chain rule.

Theorem 6.4. The solution of schemes (6.22), (6.23), (6.24) satisfies

$$\frac{1}{\Delta t}\int_{\mathbb{T}} \left(H(u^{(n+1)}, \mathcal{D}_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}^2_{X_\mathrm{p}} u^{(n+1)}) - H(u^{(n)}, \mathcal{D}_{X_\mathrm{p}} u^{(n)}, \mathcal{D}^2_{X_\mathrm{p}} u^{(n)})\right)\mathrm{d}x = 0, \qquad n = 0, 1, 2, \dots.$$

Finally, following the procedure of Phase 2, we obtain the underlying H^1-weak form; that is, we find $u(t, \cdot), p_1, q, a_1, a_2 \in H^1(\mathbb{T})$ such that for any $v_1, v_2, v_3, v_4, v_5 \in H^1(\mathbb{T})$,

$$\begin{aligned}
&(u_t, v_1) = ((p_1)_x, v_1),\\
&(p_1, v_2) = \left(-\frac{u^2}{2}, v_2\right) + \alpha(a_1, (v_2)_x) - (q, (v_2)_x),\\
&(q, v_3) = -\beta((a_2, (v_3)_x),\\
&\left(a_1, v_4\right) = \left(u_x, v_4\right),\\
&(a_2, v_5) = ((a_1)_x, v_5).
\end{aligned}$$

We now focus on numerical experiments. The implementation of schemes (6.22), (6.23), (6.24) is straightforward. Denoting the basis functions of X_p by $\psi_i(x)$ $(i = 0, 1, \dots, N-1)$, the concrete form of

the scheme is written as

$$\mathsf{A}\left(\frac{\boldsymbol{u}^{(n+1)}-\boldsymbol{u}^{(n)}}{\Delta t}\right)=\mathsf{B}\boldsymbol{p}_1^{(n+\frac{1}{2})}, \tag{6.25}$$

$$\mathsf{A}\boldsymbol{p}_1^{(n+\frac{1}{2})}=\boldsymbol{f}\Big(\boldsymbol{u}^{(n+1)},\boldsymbol{u}^{(n)},\boldsymbol{q}_1^{(n+\frac{1}{2})}\Big), \tag{6.26}$$

$$\mathsf{A}\boldsymbol{q}_1^{(n+\frac{1}{2})}=\boldsymbol{g}\big(\boldsymbol{u}^{(n+1)},\boldsymbol{u}^{(n)}\big), \tag{6.27}$$

where $\boldsymbol{u}^{(n)}:=(u_0^{(n)},u_1^{(n)},\dots,u_{N-1}^{(n)})$ are the coefficient vectors of $u^{(n)}(x)=\sum_{i=0}^{N-1}u_i^{(n)}\psi_i(x)$ (identical notation is used for $p_1^{(n+\frac{1}{2})}(x)$ and $q_1^{(n+\frac{1}{2})}(x)$), and $\boldsymbol{f}$ and $\boldsymbol{g}$ are the vectors derived from the right-hand side of (6.23) and (6.24), respectively ($\boldsymbol{f}$ is nonlinear and $\boldsymbol{g}$ is linear in $\boldsymbol{u}^{(n+1)}$). Matrix A is the mass matrix whose elements are $\mathsf{A}_{ij}=\big(\psi_i,\psi_j\big)$, while the elements of B are $\mathsf{B}_{ij}=\big((\psi_i)_x,\psi_j\big)$. Since matrix A is invertible, (6.25), (6.26), and (6.27) immediately reduce to

$$\mathsf{A}\left(\frac{\boldsymbol{u}^{(n+1)}-\boldsymbol{u}^{(n)}}{\Delta t}\right)=\mathsf{B}\mathsf{A}^{-1}\boldsymbol{f}\big(\boldsymbol{u}^{(n+1)},\boldsymbol{u}^{(n)},\mathsf{A}^{-1}\boldsymbol{g}\big((u)^{(n+1)},\boldsymbol{u}^{(n)}\big)\big).$$

Thus, we need not compute the intermediate variables $\boldsymbol{p}_1^{(n+\frac{1}{2})}$ and $\boldsymbol{q}_1^{(n+\frac{1}{2})}$, and the dimension of the nonlinear systems to be solved is reduced from $3N$ to N.

The operator $\mathcal{D}_{X_\mathrm{p}}$ is implemented as follows. Denoting $\mathcal{D}_{X_\mathrm{p}}u^{(n)}=\sum_{i=0}^{N-1}d_i^{(n)}\psi_i(x)$, the coefficient $\boldsymbol{d}^{(n)}=(d_0^{(n)},d_1^{(n)},\dots,d_{N-1}^{(n)})^\top$ is calculated by

$$\mathsf{A}\boldsymbol{d}^{(n)}=\mathsf{B}\boldsymbol{u}^{(n)},$$

which is equivalent to $\big(\mathcal{D}_X u^{(n)},\psi_i\big)=(u_x,\psi_i)$ $(i=0,1,\dots,N-1)$.

Next we check the qualitative behaviour and discrete conservation law of the numerical solution. For simplicity, we adopt a uniform mesh and P1 elements. The parameters were set to $\alpha=\beta=1$, $t=[0,400]$, $x\in[0,50]$, $\Delta x=50/101$ ($N=101$), $\Delta t=0.1$. Given that the Kawahara equation has a solitary wave solution [196]

$$u(t,x)=\frac{105\alpha^2}{169\beta}\operatorname{sech}^4\left[\frac{1}{2}\sqrt{\frac{\alpha}{13\beta}}\left(x-x_0-\frac{36\alpha^2}{169\beta}t\right)\right],\qquad x\in\mathbb{R},$$

we set the initial value to $u(0,x)=(105/169)\operatorname{sech}^4((1/2)\sqrt{1/13}(x-25))$. Figure 6.2 plots the numerical solution obtained by schemes (6.22), (6.23) and (6.24). Over an extended time, the scheme accurately captures the solitary solutions and the maximum value of the error in the discretised energy is 5.82×10^{-16} which favourably agrees with the discrete conservation law (Theorem 6.4).

Finally, we investigate the scheme from an alternative viewpoint. The following is one of the simplest H^1-weak formulations of the Kawahara equation (5.15). We find $u(t,\cdot),p,q\in H^1(\mathbb{T})$ such that for any $v_1,v_2,v_3\in H^1(\mathbb{T})$,

$$\begin{aligned}(u_t,v_1)&=\left(\frac{u^2}{2},(v_1)_x\right)+\alpha(p,(v_1)_x)-\beta(q,(v_1)_x),\\(p,v_2)&=-(u_x,(v_2)_x),\\(q,v_3)&=-(p_x,(v_3)_x).\end{aligned}$$

We consider a standard spatial discretisation of the above naive weak form (which is not conservative), and the conservative semi-discrete scheme derived by the proposed method, and time-discretise them by the fourth-order explicit Runge–Kutta method. Although the temporal discretisation destroys conservation in both formulations, the results are truly different. The time stepsize was set to $\Delta t=0.0025$

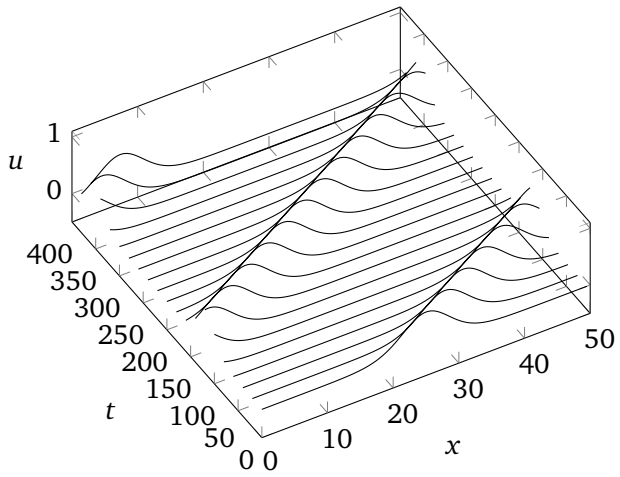


Figure 6.2: Numerical solution to the Kawahara equation solved by schemes (6.22), (6.23) and (6.24).

(other parameters were set to those of the above experiment). From Figure 6.3, we observe that numerical solution based on the conservative weak form was stable in $t \in [0, 10]$, while that based on the naive weak form was amplified uncontrollably within the first four steps. This example validates the conservative weak form itself, obtained as a byproduct of the proposed method.

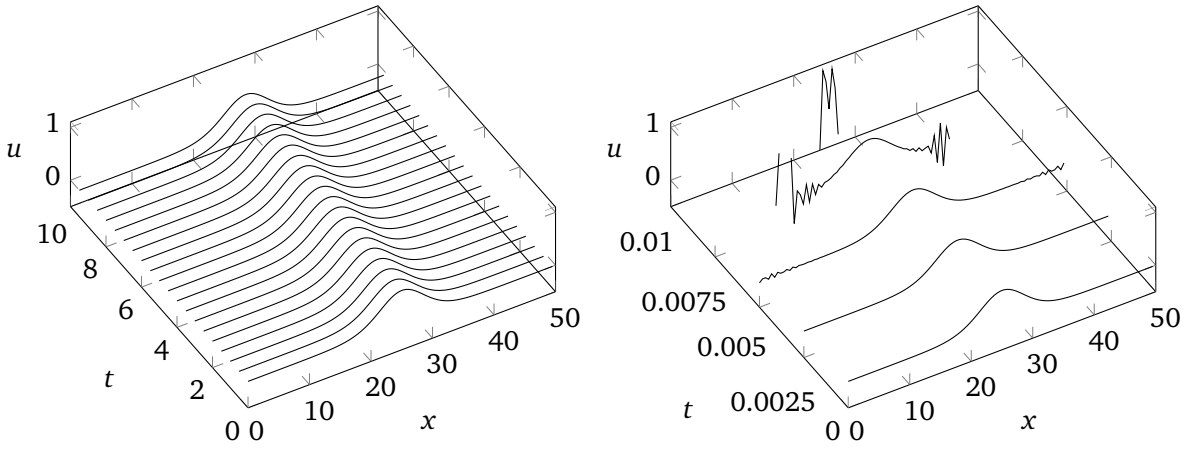


Figure 6.3: Numerical solutions to the Kawahara equation obtained by the fourth-order explicit Runge–Kutta method, based on (left) conservative weak form and (right) naive weak form.

Type 2: The Camassa–Holm equation

In this example, we derive conservative schemes for the Camassa–Holm equation (5.16), a conservative PDE of Type 2. Although one of the schemes has been previously published in [151], we here provide additional discussions, with special focus on the weak formulation.

We begin by emphasising the difficulty of finding a conservative H^1-weak form for the Camassa–Holm equation (5.16). As is well known, the Camassa–Holm equation has a characteristic peakon (peaked soliton) solution: $u(t,x) = c\exp(-|x-ct|)$ which exists in H^1 but not in C^1. One H^1-weak formulation has been established in [58] (see also [59]):

$$u_t + \frac{1}{2}\left(u^2 + \mathcal{K}\left(u^2 + \frac{u_x^2}{2}\right)\right)_x = 0, \qquad \text{where} \qquad \mathcal{K} = (1-\partial_x^2)^{-1}.$$

This formulation is inconvenient in our study since it seems to prohibit the direct establishment of a conservation law. On the other hand, Matsuo [136] proposed the following weak form: Find $m(t,\cdot), p \in H^1(\mathbb{T})$ such that for any $v_1, v_2 \in H^1(\mathbb{T})$,

$$\begin{aligned}(m_t, v_1) &= ((m\partial_x + \partial_x m)p, v_1),\\ (p, v_2) &= \left(\frac{\partial H}{\partial(\mathcal{K}m)}, \mathcal{K}v_2\right) + \left(\frac{\partial H}{\partial(\mathcal{K}m_x)}, \mathcal{K}(v_2)_x\right),\end{aligned}$$

from which the conservation law is directly obtained. One drawback of this formulation is that it captures only H^3 (when $m \in H^1$) or smoother solutions in terms of the original variable u. Note that when u is a peakon solution, m becomes a delta function.

The Camassa–Holm equation has the Hamiltonian structure

$$m_t = (m\partial_x + \partial_x m)\frac{\delta H}{\delta m},$$

where

$$H = -\frac{u^2 + u_x^2}{2}, \qquad \text{and} \qquad m = (1 - \partial_x^2)u,$$

or equivalently

$$u_t = (1-\partial_x^2)^{-1}(m\partial_x + \partial_x m)(1-\partial_x^2)^{-1}\frac{\delta H}{\delta u}. \tag{6.28}$$

Introducing an intermediate variable p, we can transform (6.28) into the system

$$\begin{aligned}(1-\partial_x^2)u_t &= (m\partial_x + \partial_x m)p,\\ (1-\partial_x^2)p &= \frac{\delta H}{\delta u}.\end{aligned} \tag{6.29}$$

First, we consider the following formal weak form: Find u, p such that for any v_1, v_2,

$$\left((1-\partial_x^2)u_t, v_1\right) = ((m\partial_x + \partial_x m)p, v_1), \tag{6.30}$$

$$\left((1-\partial_x^2)p, v_2\right) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right). \tag{6.31}$$

The conservation law is explicitly obtained by formal calculation:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u,u_x)\,\mathrm{d}x &= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial u_x}, u_{xt}\right) = \left((1-\partial_x^2)p, u_t\right)\\ &= \left(p, (1-\partial_x^2)u_t\right) = ((m\partial_x + \partial_x m)p, p) = 0.\end{aligned}$$

The first equality is an application of the chain rule. The second equality follows from (6.31) with $v_2 = u_t$, the third from the symmetry of $(1-\partial_x^2)$, and the fourth from (6.30) with $v_1 = p$. The last derives from the skew-symmetry of $(m\partial_x + \partial_x m)$.

Remark 6.9. The transformation from (6.28) to (6.29) is automatic in the following sense. In performing the transformation, we note that

- skew-symmetric operators should be retained (in this case, $(m\partial_x + \partial_x m)$);
- variational derivatives (such as the second equation of (6.29)) require separate treatment.

Then we can easily find a formally conservative system. Such a system is usually nonunique; for instance, another formally conservative system for the Camassa–Holm equation is

$$u_t = (1-\partial_x^2)^{-1}(m\partial_x + \partial_x m)(1-\partial_x^2)^{-1}p,$$
$$p = \frac{\delta H}{\delta u}.$$

The subsequent procedure is applicable to all of these systems.

Second, we derive a semi-discrete scheme from the L^2-projection operators. Suppose that $u(0,\cdot) \in X_{\mathrm{p}}$ is given. We find $u(t,\cdot), p(t,\cdot) \in X_{\mathrm{p}}$ such that for any $v_1, v_2 \in X_{\mathrm{p}}$,

$$\Big((1-(\mathcal{D}_{X_{\mathrm{p}}})^2)u_t, v_1\Big) = ((m\partial_x + \partial_x m)p, v_1),$$
$$\Big((1-(\mathcal{D}_{X_{\mathrm{p}}})^2)p, v_2\Big) = \left(\frac{\partial H}{\partial u}, v_2\right) + \left(\frac{\partial H}{\partial u_x}, (v_2)_x\right),$$

where $m = (1-(\mathcal{D}_{X_{\mathrm{p}}})^2)u$. This semi-discrete scheme is consistent in $X_{\mathrm{p}} \subset H^1(\mathbb{T})$, and has the rigorous conservation property

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u,u_x)\,\mathrm{d}x = 0.$$

Remark 6.10. If the rules of the proposed method are strictly obeyed, $(m\partial_x + \partial_x m)$ should be replaced with $(m\mathcal{D}_{X_{\mathrm{p}}} + \mathcal{D}_{X_{\mathrm{p}}} m)$. However, as mentioned in Remark 6.7, we can omit the replacement of ∂_x because in this case the operator $(m\partial_x + \partial_x m)$ with $m = (1-\mathcal{D}_{X_{\mathrm{p}}}^2)u$ is valid in H^1. The above semi-discrete scheme is identical to that proposed in our recent report [151].

Third, we time-discretise the above semi-discrete scheme to obtain the following fully discrete scheme. Suppose that $u^{(0)} \in X_{\mathrm{p}}$ is given. We find $u^{(n+1)}, p^{(n+\frac{1}{2})} \in X_{\mathrm{p}}$ $(n = 0,1,\dots)$ such that for any $v_1, v_2 \in X_{\mathrm{p}}$,

$$\left(\Big(1-(\mathcal{D}_{X_{\mathrm{p}}})^2\Big)\frac{u^{(n+1)}-u^{(n)}}{\Delta t}, v_1\right) = \Big((m^{(n+\frac{1}{2})}\partial_x + \partial_x m^{(n+\frac{1}{2})})p^{(n+\frac{1}{2})}, v_1\Big), \tag{6.32}$$

$$\Big(\Big(1-(\mathcal{D}_{X_{\mathrm{p}}})^2\Big)p^{(n+\frac{1}{2})}, v_2\Big) = \left(\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)}, u^{(n)})}, v_2\right) + \left(\frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})}, (v_2)_x\right), \tag{6.33}$$

where $m^{(n+\frac{1}{2})} = (1-(\mathcal{D}_{X_{\mathrm{p}}})^2)(u^{(n+1)} + u^{(n)})/2$,

$$\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)}, u^{(n)})} = -\frac{u^{(n+1)} + u^{(n)}}{2}, \qquad \frac{\partial H_{\mathrm{d}}}{\partial(u_x^{(n+1)}, u_x^{(n)})} = -\frac{u_x^{(n+1)} + u_x^{(n)}}{2}.$$

Theorem 6.5. The solution of schemes (6.32), (6.33) satisfies

$$\frac{1}{\Delta t}\int_{\mathbb{T}} \big(H(u^{(n+1)}, u_x^{(n+1)}) - H(u^{(n)}, u_x^{(n)})\big)\,\mathrm{d}x = 0, \qquad n = 0,1,2,\dots.$$

Finally, following the procedure of Phase 2, we obtain the underlying H^1-weak form. We find

$u, m, p, q_1, q_2, q_3 \in H^1(\mathbb{T})$ such that for any $v_1, \dots, v_6 \in H^1(\mathbb{T})$,

$$\begin{aligned}
&(m_t, v_1) = ((m\partial_x + \partial_x m)p, v_1),\\
&(m, v_2) = (u, v_2) + (q_1, (v_2)_x),\\
&(q_1, v_3) = (u_x, v_3),\\
&(q_2, v_4) = \left(\frac{\partial H}{\partial u}, v_4\right) + \left(\frac{\partial H}{\partial u_x}, (v_4)_x\right),\\
&(q_2, v_5) = (p, v_5) + (q_3, (v_5)_x),\\
&(q_3, v_6) = (p_x, v_6).
\end{aligned}$$

The solution of this weak form is conservative as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{T}} H(u, u_x)\,\mathrm{d}x = 0.$$

6.3 Extension to multidimensional problems

In this section, the proposed method is extended to multidimensional cases. The extension is illustrated by example. Consider a dissipative equation of the form

$$\frac{\partial u}{\partial t} = (-1)^{s+1}\triangle^s \frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, \nabla u, \triangle u), \tag{6.34}$$

where the variational derivative in two or three dimensions is defined by

$$\frac{\delta H}{\delta u} := \frac{\partial H}{\partial u} - \nabla \cdot \frac{\partial H}{\partial \nabla u} + \triangle \frac{\partial H}{\partial \triangle u}.$$

We first introduce the notation of the L^2-projection operators in higher-dimensions, and demonstrate their properties. Next we derive dissipative schemes for (6.34) with $s = 0$. As an example, we adopt the two-dimensional Swift–Hohenberg (2D-SH) equation

$$u_t = -u^3 + 2u - 2\nabla u - \triangle^2 u,$$

on the torus $\mathbb{T}^2$, whose energy functional is $H(u, \nabla u, \triangle u) = u^4/4 - u^2 - |\nabla u|^2 + (\triangle u)^2/2$. To conserve space, we show only a formal weak form and the resulting scheme, but the underlying weak form can be also derived by the procedure of Phase 2.

6.3.1 L^2-projection operators in multidimensional cases

We define L^2-projection operators $\mathcal{P}_X : L^2 \to X \subseteq H^1(\Omega) \subset L^2(\Omega)$ satisfying

$$(\mathcal{P}_X u, v) = (u, v)$$

for any $v \in X$, and $\mathcal{P}_{\boldsymbol{X}} : \boldsymbol{L}^2(\Omega) \to \boldsymbol{X} \subseteq \boldsymbol{H}^1(\Omega) \subset \boldsymbol{L}^2(\Omega)$ satisfying

$$(\mathcal{P}_{\boldsymbol{X}} \boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u}, \boldsymbol{v})$$

for any $\boldsymbol{v} \in \boldsymbol{X}$. We also denote $\mathcal{P}_{\boldsymbol{X}} \nabla u$ and $\mathcal{P}_X \nabla \cdot \boldsymbol{u}$ by $\mathcal{D}_{\boldsymbol{X}} u$ and $\mathcal{D}_X \boldsymbol{u}$. That is, $\mathcal{D}_{\boldsymbol{X}} := \mathcal{P}_{\boldsymbol{X}} \nabla : H^1(\Omega) \to \boldsymbol{X}$ and $\mathcal{D}_X := \mathcal{P}_X \nabla \cdot : \boldsymbol{H}^1(\Omega) \to X$. The following formulas for these operators are analogous to Lemma 6.1 and 6.1, and are straightforward.

Lemma 6.3. For any $u \in H^1(\Omega)$ and $\boldsymbol{v} \in \boldsymbol{X}$, the following holds

$$(\mathcal{D}_{\boldsymbol{X}} u, \boldsymbol{v}) = (\nabla u, \boldsymbol{v}),$$

and for any $\boldsymbol{u} \in \boldsymbol{H}^1(\Omega)$ and $v \in X$, the following holds

$$(\mathcal{D}_X \boldsymbol{u}, v) = (\nabla \cdot \boldsymbol{u}, v).$$

Corollary 6.2. For any $u \in X$ and $\boldsymbol{v} \in \boldsymbol{X}$ such that $\int_\Gamma u(\boldsymbol{n} \cdot \boldsymbol{v})\,\mathrm{d}\Gamma = 0$, we have

$$(\mathcal{D}_{\boldsymbol{X}} u, \boldsymbol{v}) = -(u, \mathcal{D}_X \boldsymbol{v}). \tag{6.35}$$

For any $u \in X$ and $v \in X$ such that $\int_\Gamma (\boldsymbol{n} \cdot \mathcal{D}_{\boldsymbol{X}} u) v\,\mathrm{d}\Gamma = \int_\Gamma u(\boldsymbol{n} \cdot \mathcal{D}_{\boldsymbol{X}} v)\,\mathrm{d}\Gamma = 0$, we have

$$(\mathcal{D}_X \mathcal{D}_{\boldsymbol{X}} u, v) = (u, \mathcal{D}_X \mathcal{D}_{\boldsymbol{X}} v). \tag{6.36}$$

For any $\boldsymbol{u} \in \boldsymbol{X}$ and $\boldsymbol{v} \in \boldsymbol{X}$ such that $\int_\Gamma (\mathcal{D}_X \boldsymbol{u})(\boldsymbol{n} \cdot \boldsymbol{v})\,\mathrm{d}\Gamma = \int_\Gamma (\boldsymbol{n} \cdot \boldsymbol{u})(\mathcal{D}_X \boldsymbol{v})\,\mathrm{d}\Gamma = 0$, we have

$$(\mathcal{D}_X \mathcal{D}_{\boldsymbol{X}} u, v) = (u, \mathcal{D}_X \mathcal{D}_{\boldsymbol{X}} v). \tag{6.37}$$

Proof. Eq. (6.35) directly follows from the Green theorem (6.1). Eqs. (6.36) and (6.37) immediately follow from (6.35). □

We can also define operators corresponding to $\mathcal{P}_{X(Y)}$ and $\mathcal{D}_{X(Y)}$, but these are omitted here.

6.3.2 Application to the 2D-SH equation

We now derive a dissipative scheme for the 2D-SH equation, assuming the periodic boundary conditions for simplicity. We first show a formal weak form and its fully discrete scheme, and then plot the numerical results.

We begin with the following formal weak form. We find u and $\boldsymbol{q}$, such that for any v_1 and $\boldsymbol{v}_2$,

$$\begin{aligned}
(u_t, v_1) &= -\left(\frac{\partial H}{\partial u}, v_1\right) - \left(\frac{\partial H}{\partial \nabla u}, \nabla v_1\right) + (\boldsymbol{q}, \nabla v_1), \\
(\boldsymbol{q}, \boldsymbol{v}_2) &= -\left(\frac{\partial H}{\partial \triangle u}, \nabla \cdot \boldsymbol{v}_2\right).
\end{aligned}$$

Since this formal weak form is completely analogous to Formal weak form 1, the subsequent procedures are straightforward, and we simply state the fully discrete scheme. Suppose that $u^{(0)}$ is given in X_p. We find $u^{(n+1)} \in X_\mathrm{p}$ and $\boldsymbol{q}^{(n+\frac{1}{2})} \in \boldsymbol{X}_\mathrm{p}$ $(n = 0, 1, \dots)$ such that for any $v_1 \in X_\mathrm{p}$ and $\boldsymbol{v}_2 \in \boldsymbol{X}_\mathrm{p}$,

$$\begin{aligned}
\left(\frac{u^{(n+1)} - u^{(n)}}{\Delta t}, v_1\right) = &-\left(\frac{\partial H_\mathrm{d}}{\partial (u^{(n+1)}, u^{(n)})}, v_1\right) \\
&- \left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H_\mathrm{d}}{\partial (\mathcal{D}_{X_\mathrm{p}} u^{(n+1)}, \mathcal{D}_{X_\mathrm{p}} u^{(n)})}, \nabla v_1\right) + \left(\boldsymbol{q}^{(n+\frac{1}{2})}, \nabla v_1\right),
\end{aligned} \tag{6.38}$$

$$\left(\boldsymbol{q}^{(n+\frac{1}{2})}, \boldsymbol{v}_2\right) = -\left(\mathcal{P}_{X_\mathrm{p}} \frac{\partial H_\mathrm{d}}{\partial (\mathcal{D}_{X_\mathrm{p}} \mathcal{D}_{\boldsymbol{X}_\mathrm{p}} u^{(n+1)}, \mathcal{D}_{X_\mathrm{p}} \mathcal{D}_{\boldsymbol{X}_\mathrm{p}} u^{(n)})}, \nabla \cdot \boldsymbol{v}_2\right), \tag{6.39}$$

where

$$\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)},u^{(n)})}=\frac{((u^{(n+1)})^2+(u^{(n)})^2)(u^{(n+1)}+u^{(n)})}{4}-(u^{(n+1)}+u^{(n)}),$$
$$\frac{\partial H_{\mathrm{d}}}{\partial(\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)},\mathcal{D}_{X_{\mathrm{p}}}u^{(n)})}=-(\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)}+\mathcal{D}_{X_{\mathrm{p}}}u^{(n)}),$$
$$\frac{\partial H_{\mathrm{d}}}{\partial(\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)},\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n)})}=\frac{\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)}+\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n)}}{2},$$

which correspond to $\partial H/\partial u=u^3-2u$, $\partial H/\partial\nabla u=-2\nabla u$, and $\partial H/\partial\triangle u=\triangle u$.

Theorem 6.6. The solution of schemes (6.38) and (6.39) satisfies

$$\frac{1}{\Delta t}\int_\Omega\Big(H(u^{(n+1)},\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)},\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n+1)})$$
$$-H(u^{(n)},\mathcal{D}_{X_{\mathrm{p}}}u^{(n)},\mathcal{D}_{X_{\mathrm{p}}}\mathcal{D}_{X_{\mathrm{p}}}u^{(n)})\Big)\,\mathrm{d}x\le 0,\qquad n=0,1,2,\dots.$$

6.4 Extension to local discontinuous Galerkin framework

We established the general Galerkin framework of energy-preserving/dissipative methods in the previous sections. The framework was constructed by assuming P1 elements. Therefore, the implementation is easy, while the accuracy is limited. The aim of this section is to construct a spatially high-order Galerkin method which is still easy to implement. For this aim, we further extend the energy-preserving/dissipative method to the local discontinuous Galerkin framework.

The discontinuous Galerkin (DG) method is a variant of finite element method that uses discontinuous piecewise polynomial spaces for test and trial functions. It can be regarded as something between finite element and finite volume methods, and thanks to the discontinuity of functions, it has favourable features that it is easy to increase the order of accuracy, and also that the resulting schemes are highly parallelisable when schemes are explicit. The DG method was first introduced by Reed–Hill for solving hyperbolic equations [169]. It was then extended by Bassi–Rebay [8] for an elliptic problem so that higher-order derivatives can be also handled. Encouraged by this success, Cockburn–Shu [52] developed a generalisation called local discontinuous Galerkin (LDG) method. The basic idea of the LDG method is to rewrite a higher-order differential equation into a system of first-order equations by employing intermediate variables. History and further information of the DG method can be found in Cockburn et al. [51]. The DG method has various examples. Below we list limited examples. Yan–Shu [212] applied LDG method to the KdV equation. DG method has been also applied to other nonlinear wave equations such as the Camassa–Holm equation [205] and nonlinear Schrödinger equation [204]. Xia–Xu–Shu applied the LDG method to the Cahn–Hilliard equation [202]. DG methods have been also combined with structure-preserving methods. Xing–Chou–Shu [203] proposed an energy-preserving LDG scheme for linear wave equations and gave an error estimate. Bona–Chen–Karakashian–Xing [15] derived a quadratic invariant preserving DG scheme for the generalised KdV equation. A similar study has been made by Yi–Huang–Liu [214] where a variant of DG method, called direct DG method, is employed.

In this section, we show that we can automatically construct energy-preserving/dissipative LDG schemes for wide variety of variational PDEs. To clarify our idea, we only consider conservative PDEs of the form

$$u_t=\partial_x\frac{\delta\mathcal{H}}{\delta u},\qquad H=H(u,u_x),\tag{6.40}$$

and dissipative PDEs of the form

$$u_t = \partial_x^2 \frac{\delta \mathcal{H}}{\delta u}, \qquad H = H(u, u_x). \tag{6.41}$$

We further make an assumption that $H(u, u_x)$ is separable: $H(u, u_x) = H_1(u) + H_2(u_x)$ for some functions H_1 and H_2, and H_2 is a quadratic function. This greatly simplifies the discussion, and still covers many PDEs. Below we show weak forms of the above PDEs. The weak forms, derived by the technique established in the previous sections, are based on systems of first order equations.

Weak form 3 (Energy-preserving weak form of (6.40))**.** Find $u(t,\cdot), p, q \in H^1(0,L)$ such that, for any $v_1, v_2, v_3 \in H^1(0,L)$,

$$(u_t, v_1) = (p_x, v_1), \tag{6.42}$$

$$(p, v_2) = \left(\frac{\partial H}{\partial u}, v_2\right) - \left(\partial_x \frac{\partial H}{\partial q}, v_2\right), \tag{6.43}$$

$$(q, v_3) = (u_x, v_3). \tag{6.44}$$

We usually restrict function spaces to appropriate subspaces of $H^1(0,L)$ corresponding to boundary conditions. Nevertheless to make the discussion in the next subsection clear and avoid confusing notation, we leave this issue to later consideration, and simply assume the existence of the solution.

Theorem 6.7 (Energy-preservation of Weak form 3)**.** Assume that $u_t \in H^1(0,L)$, and the boundary conditions satisfy

$$\left[\frac{\partial H}{\partial q} u_t\right]_0^L = 0, \qquad [p^2]_0^L = 0.$$

Then the solution of Weak form 3 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_0^L H(u,q)\,\mathrm{d}x = 0.$$

Proof.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} \int_0^L H(u,q)\,\mathrm{d}x &= \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial q}, q_t\right) = \left(\frac{\partial H}{\partial u}, u_t\right) + \left(\frac{\partial H}{\partial q}, u_{xt}\right) \\ &= \left(\frac{\partial H}{\partial u}, u_t\right) - \left(\partial_x \frac{\partial H}{\partial q}, u_t\right) + \left[\frac{\partial H}{\partial q} u_t\right]_0^L = (p, u_t) = (p, p_x) = \frac{1}{2}[p^2]_0^L = 0.\end{aligned}$$

The first equality is a simple application of the chain rule. Temporally differentiating (6.44) and substituting $v_3 = \partial H/\partial q$, we obtain the second equality. This procedure is allowed by the assumption that $\partial H/\partial q \propto q$ belongs to the same space as q. The third equality is obtained by integration-by-parts. The fourth and fifth equalities follow from (6.43) with $v_2 = u_t$ and (6.42) with $v_1 = p$, respectively. In the calculation, the boundary terms are eliminated due to the boundary conditions. □

Remark 6.11. It is also possible to consider more general PDEs. For example, if we drop the restriction on $H(u, u_x)$ (i.e., $H(u, u_x)$ is not necessarily separable and quadratic with respect to u_x), we find the following weak form: Find $u(t,\cdot), p, q, w \in H^1(0,L)$ such that, for any $v_1, v_2, v_3, v_4 \in H^1(0,L)$,

$$\begin{aligned}(u_t, v_1) &= (p_x, v_1), \\ (p, v_2) &= \left(\frac{\partial H}{\partial u}, v_2\right) - (w_x, v_2), \\ (w, v_3) &= \left(\frac{\partial H}{\partial q}, v_3\right), \\ (q, v_4) &= (u_x, v_4).\end{aligned}$$

Then we can carry out the same procedure below, but the discussion would become slightly cumbersome due to the additional intermediate variable w. Note that the above weak form reduces to Weak form 3 when $H(u,u_x)$ is separable: $H(u,u_x)=H_1(u)+H_2(u_x)$, and H_2 is a quadratic function.

Remark 6.12. A key device for obtaining intended weak forms is to use the framework of the previous sections so that the partial derivative ∂_x does not operate on the test functions. For example, for the equation (6.40), we start with the following formal weak form

$$\begin{aligned}&(u_t,v_1)=(p_x,v_1),\\&(p,v_2)=\left(\frac{\partial H}{\partial u},v_2\right)-\left(\partial_x\frac{\partial H}{\partial u_x},v_2\right).\end{aligned}$$

By defining appropriate L^2-projection operators $\mathcal{P}_X$ and $\mathcal{D}_X$, we obtain the semi-discrete scheme

$$\begin{aligned}&(u_t,v_1)=(p_x,v_1),\\&(p,v_2)=\left(\frac{\partial H}{\partial u},v_2\right)-\left(\partial_x\left(\mathcal{P}_X\frac{\partial H}{\partial(\mathcal{D}_X u)}\right),v_2\right).\end{aligned}$$

Finally, by following the procedures of Phase 2 (i.e., eliminating the projection operators), we have the above weak formulation.

Weak form 4 (Energy-dissipative weak form of (6.41))**.** Find $u(t,\cdot),p,q,r\in H^1(0,L)$ such that, for any $v_1,v_2,v_3,v_4\in H^1(0,L)$,

$$(u_t,v_1)=(r_x,v_1), \tag{6.45}$$
$$(r,v_2)=(p_x,v_2), \tag{6.46}$$
$$(p,v_3)=\left(\frac{\partial H}{\partial u},v_3\right)-\left(\partial_x\frac{\partial H}{\partial q},v_3\right), \tag{6.47}$$
$$\big(q,v_4\big)=\big(u_x,v_4\big). \tag{6.48}$$

Theorem 6.8 (Energy-dissipation of Weak form 4)**.** Assume that $u_t\in H^1(0,L)$, and the boundary conditions satisfy

$$\left[\frac{\partial H}{\partial q}u_t\right]_0^L=0,\qquad [rp]_0^L=0.$$

Then the solution of Weak form 4 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x\le 0.$$

Proof.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x&=\left(\frac{\partial H}{\partial u},u_t\right)+\left(\frac{\partial H}{\partial q},q_t\right)=\left(\frac{\partial H}{\partial u},u_t\right)+\left(\frac{\partial H}{\partial q},u_{xt}\right)\\&=\left(\frac{\partial H}{\partial u},u_t\right)-\left(\partial_x\frac{\partial H}{\partial q},u_t\right)+\left[\frac{\partial H}{\partial q}u_t\right]_0^L=(p,u_t)=(r_x,p)\\&=-(r,p_x)+[rp]_0^L=-(r,r)=-\|r\|^2_{L^2(0,L)}\le 0.\end{aligned}$$

The second equality is obtained by (6.48) with $v_4=\partial H/\partial q$. The fourth, fifth and seventh equalities follow from (6.47) with $v_3=u_t$, (6.45) with $v_1=p$ and (6.46) with $v_2=r$, respectively. □

6.4.1 Energy-preserving/dissipative LDG method

We here propose a new method to derive energy-preserving/dissipative LDG schemes.

We divide the computational domain $[0, L]$ into N intervals

$$0 = x_{1/2} < \cdots < x_{j-1/2} < x_{j+1/2} < \cdots < x_{N+1/2} = L.$$

We denote the computational cell by $I_j = (x_{j-1/2}, x_{j+1/2})$ for $j = 1, \ldots, N$. We denote by $u^+_{j+1/2}$ and $u^-_{j+1/2}$ the values of u at $x_{j+1/2}$, from the right cell I_{j+1} and from the left cell I_j (see Figure 6.4). This rule applies also to other variables and functions. We define the piecewise polynomial space V_h as the space of polynomials of degree up to k in each cell I_j, i.e.,

$$V_h = \{v : v|_{I_j} \in P^k(I_j), \qquad j = 1, \ldots, N\}.$$

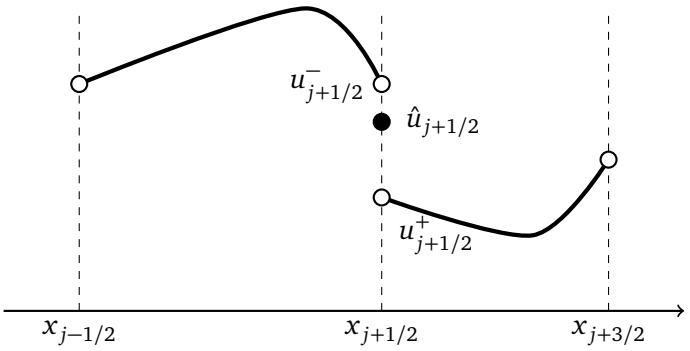


Figure 6.4: Notation in LDG methods.

Conservative cases

Here, we shall derive an energy-preserving LDG scheme. We first show a semi-discrete scheme and then summarise the essential idea of our method. Finally, we derive a fully discrete scheme. We start the derivation with the following abstract form of a semi-discrete LDG scheme, which is obtained from Weak form Weak form 3.

Semi-discrete scheme 4. Find $u(t, \cdot), p, q \in V_h$ such that, for any $v_1, v_2, v_3 \in V_h$ and for $j = 1, \ldots, N$,

$$(u_t, v_1)_{I_j} = -(p, (v_1)_x)_{I_j} + [\hat{p} v_1]_{I_j}, \tag{6.49}$$

$$(p, v_2)_{I_j} = \left(\frac{\partial H}{\partial u}, v_2\right)_{I_j} + \left(\frac{\partial H}{\partial q}, (v_2)_x\right)_{I_j} - \left[\widehat{\frac{\partial H}{\partial q}} v_2\right]_{I_j}, \tag{6.50}$$

$$(q, v_3)_{I_j} = -(u, (v_3)_x)_{I_j} + [\hat{u} v_3]_{I_j}, \tag{6.51}$$

where $[\hat{f} w]_{I_j} = \hat{f}_{j+1/2} w_{j+1/2} - \hat{f}_{j-1/2} w_{j-1/2}$.

The "hat" terms, called numerical fluxes, result from integration-by-parts in each cell, and are single valued functions defined on the edges (see Figure 6.4). In the standard LDG theory, these terms are introduced to ensure the numerical stability and reflect boundary conditions. Here we show that there is another choice such that the semi-discrete scheme become energy-preserving. In what follows, we call fluxes at $x = x_{1/2},\ x_{N+1/2}$ boundary fluxes, and others internal fluxes.

We assume that internal fluxes are given by, for $j = 1, \dots, N-1$,

$$\hat{p}_{j+1/2} = \frac{1}{2}\Big(p^+_{j+1/2} + p^-_{j+1/2}\Big), \tag{6.52}$$

$$\widehat{\frac{\partial H}{\partial q}}_{j+1/2} = \lambda \frac{\partial H}{\partial q}^{+}_{j+1/2} + (1-\lambda)\frac{\partial H}{\partial q}^{-}_{j+1/2}, \tag{6.53}$$

$$\hat{u}_{j+1/2} = (1-\lambda)u^+_{j+1/2} + \lambda u^-_{j+1/2}, \tag{6.54}$$

with a real parameter λ, and boundary fluxes are set to satisfy

$$\left(\frac{1}{2}p^-_{N+1/2} - \hat{p}_{N+1/2}\right)p^-_{N+1/2} - \left(\frac{1}{2}p^+_{1/2} - \hat{p}_{1/2}\right)p^+_{1/2} = 0, \tag{6.55}$$

$$\begin{aligned}\left(u^-_{N+1/2}\right)_t \frac{\partial H}{\partial q}^{-}_{N+1/2} - \left(\hat{u}_{N+1/2}\right)_t \frac{\partial H}{\partial q}^{-}_{N+1/2} - \left(u^-_{N+1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{N+1/2} \\ - \left(u^+_{1/2}\right)_t \frac{\partial H}{\partial q}^{+}_{1/2} + \left(\hat{u}_{1/2}\right)_t \frac{\partial H}{\partial q}^{+}_{1/2} + \left(u^+_{1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{1/2} = 0.\end{aligned} \tag{6.56}$$

Obviously, the conditions (6.55) and (6.56) corresponds to $[p^2]_0^L = 0$ and $\left[\frac{\partial H}{\partial q}u_t\right]_0^L = 0$, respectively. We will discuss the derivation of the above energy-preserving fluxes after seeing the following theorem and its proof.

Theorem 6.9. Assume that the fluxes are set to (6.52), (6.53), (6.54), and set such that (6.55), (6.56) hold. Then the solution of Semi-discrete scheme 4 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x = 0.$$

Proof. First, we note that for Semi-discrete scheme 4 the following holds.

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x = -\Theta_{p^2} - \Theta_{uq},$$

where

$$\Theta_{p^2} = \sum_{j=1}^{N}\left[\frac{1}{2}p^2 - \hat{p}p\right]_{I_j}, \qquad \Theta_{uq} = \sum_{j=1}^{N}\left[u_t\frac{\partial H}{\partial q} - \hat{u}_t\frac{\partial H}{\partial q} - u_t\widehat{\frac{\partial H}{\partial q}}\right]_{I_j},$$

independently of the choice of fluxes. This can be checked as follows.

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x \\
&=\sum_{j=1}^{N}\left\{\left(\frac{\partial H}{\partial u},u_t\right)_{I_j}+\left(\frac{\partial H}{\partial q},q_t\right)_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{\left(\frac{\partial H}{\partial u},u_t\right)_{I_j}-\left(\left(\frac{\partial H}{\partial q}\right)_x,u_t\right)_{I_j}+\left[\frac{\partial G}{\partial q}\hat{u}_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{(p,u_t)_{I_j}-\left(\frac{\partial H}{\partial q},u_{xt}\right)_{I_j}-\left(\left(\frac{\partial H}{\partial q}\right)_x,u_t\right)+\left[\frac{\partial H}{\partial q}\hat{u}_t+\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{-(p,p_x)_{I_j}+[\hat{p}p]_{I_j}-\left[\frac{\partial H}{\partial q}u_t-\frac{\partial H}{\partial q}\hat{u}_t-\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{-\left[\frac{1}{2}p^2-\hat{p}p\right]_{I_j}-\left[\frac{\partial H}{\partial q}u_t-\frac{\partial H}{\partial q}\hat{u}_t-\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=-\Theta_{p^2}-\Theta_{uq}.
\end{aligned}
\tag{6.57}
$$

This calculation is quite similar to that in the proof of Theorem 6.8. The first equality is a simple application of the chain rule. The second follows from (6.51) with $v_3=\partial H/\partial q$. The third and fourth are obtained from (6.50) with $v_2=u_t$ and (6.49) with $v_1=p$, respectively.

Next, we show that $\Theta_{p^2}=\Theta_{uq}=0$. Since Θ_{p^2} is rewritten as

$$
\begin{aligned}
\Theta_{p^2}&=\sum_{j=1}^{N}\left\{\frac{1}{2}\left(p^-_{j+1/2}\right)^2-\frac{1}{2}\left(p^+_{j-1/2}\right)^2-\hat{p}_{j+1/2}p^-_{j+1/2}+\hat{p}_{j-1/2}p^+_{j-1/2}\right\} \\
&=\sum_{j=1}^{N-1}\left\{\frac{1}{2}\left(\left(p^-_{j+1/2}\right)^2-\left(p^+_{j+1/2}\right)^2\right)-\hat{p}_{j+1/2}\left(p^-_{j+1/2}-p^+_{j+1/2}\right)\right\} \\
&\quad+\frac{1}{2}\left(p^-_{N+1/2}\right)^2-\hat{p}_{N+1/2}p^-_{N+1/2}-\frac{1}{2}\left(p^+_{1/2}\right)^2+\hat{p}_{1/2}p^+_{1/2} \\
&=\sum_{j=1}^{N-1}\left\{\frac{1}{2}\left(p^-_{j+1/2}+p^+_{j+1/2}\right)-\hat{p}_{j+1/2}\right\}\left(p^-_{j+1/2}-p^+_{j+1/2}\right) \\
&\quad+\left(\frac{1}{2}p^-_{N+1/2}-\hat{p}_{N+1/2}\right)p^-_{N+1/2}-\left(\frac{1}{2}p^+_{1/2}-\hat{p}_{1/2}\right)p^+_{1/2},
\end{aligned}
$$

$\Theta_{p^2} = 0$ holds under the assumptions (6.52) and (6.55). Similarly, since Θ_{uq} is rewritten as

$$\begin{aligned}
\Theta_{uq} &= \sum_{j=1}^{N} \Bigg\{ \left(u^-_{j+1/2}\right)_t \frac{\partial H^-}{\partial q}_{j+1/2} - \left(\hat{u}_{j+1/2}\right)_t \frac{\partial H^-}{\partial q}_{j+1/2} - \left(u^-_{j+1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{j+1/2} \\
&\quad - \left(u^+_{j-1/2}\right)_t \frac{\partial H^+}{\partial q}_{j-1/2} + \left(\hat{u}_{j-1/2}\right)_t \frac{\partial H^+}{\partial q}_{j-1/2} + \left(u^+_{j-1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{j-1/2} \Bigg\} \\
&= \sum_{j=1}^{N-1} \Bigg\{ \left(\left(u^-_{j+1/2}\right)_t \frac{\partial H^-}{\partial q}_{j+1/2} - \left(u^+_{j+1/2}\right)_t \frac{\partial H^+}{\partial q}_{j+1/2} \right) \\
&\quad - \left(\hat{u}_{j+1/2}\right)_t \left(\frac{\partial H^-}{\partial q}_{j+1/2} - \frac{\partial H^+}{\partial q}_{j+1/2} \right) - \left(\left(u^-_{j+1/2}\right)_t - \left(u^+_{j+1/2}\right)_t \right) \widehat{\frac{\partial H}{\partial q}}_{j+1/2} \Bigg\} \\
&\quad + \left(u^-_{N+1/2}\right)_t \frac{\partial H^-}{\partial q}_{N+1/2} - \left(\hat{u}_{N+1/2}\right)_t \frac{\partial H^-}{\partial q}_{N+1/2} - \left(u^-_{N+1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{N+1/2} \\
&\quad - \left(u^+_{1/2}\right)_t \frac{\partial H^+}{\partial q}_{1/2} + \left(\hat{u}_{1/2}\right)_t \frac{\partial H^+}{\partial q}_{1/2} + \left(u^+_{1/2}\right)_t \widehat{\frac{\partial H}{\partial q}}_{1/2},
\end{aligned}$$

$\Theta_{uq} = 0$ holds under the assumptions (6.53), (6.54) and (6.56). This completes the proof. □

Here we summarise the procedure to find energy-preserving fluxes. Note that the calculation (6.57) is standard in the LDG context, whereas the terms Θ_{p^2} and Θ_{uq} are intrinsic to the *discontinuous* case—i.e., they essentially do not appear in the standard *continuous* Galerkin context. Thus, it is natural to demand that these terms vanish $\Theta_{p^2} = \Theta_{uq} = 0$ by choosing special fluxes. In order to find such fluxes, we separate Θ_{p^2} and Θ_{uq} into the internal and boundary terms, and first choose internal fluxes such that the terms Θ_{p^2} and Θ_{uq} are cancelled out in internal edges. Then we confirm the remaining terms successfully correspond to the original boundary conditions, so that we can set appropriate discrete boundary conditions. Note also that, in the standard DG, the strategy is different in that they are set such that $\Theta_{p^2}, \Theta_{uq} \geq 0$ hold, which often implies "energy stability."

Remark 6.13. In this and next remarks, we mention the treatment of boundary conditions (which as described before we basically ignore in the main text). First, let us consider the periodic boundary conditions. In this case, obviously the internal fluxes (6.52), (6.53) and (6.54) can be used throughout the domain, only with the small modification for periodicity: $u_{1/2} = u_{N+1/2}$, $p_{1/2} = p_{N+1/2}$, $q_{1/2} = q_{N+1/2}$. Then the boundary conditions (6.55) and (6.56) are automatically satisfied. Next, let us consider the Dirichlet boundary condition $u|_{x=0} = u|_{x=L} = 0$. To simplify the presentation, let us in particular consider the case that $\partial H/\partial q = q$; in this case the numerical flux $\widehat{\partial h}/\partial q$ is simplified as $\hat{q}$. Then we just take $\hat{u}_{1/2} = \hat{u}_{N+1/2} = 0$, $\hat{p}_{1/2} = \frac{1}{2}p^+_{1/2}, \hat{p}_{N+1/2} = \frac{1}{2}p^-_{N+1/2}$, $\hat{q}_{1/2} = q^+_{1/2}, \hat{q}_{N+1/2} = q^-_{N+1/2}$. The first one corresponds to the Dirichlet condition, and the rest are set such that (6.55) and (6.56) are satisfied.

Now we are in a position to derive a fully-discrete scheme in which the discrete partial derivatives are used (recall that the discrete partial derivatives are defined in Definition 6.1).

Scheme 4. Find $u^{(n+1)}, p^{(n+1/2)}, q^{(n+1/2)} \in V_h$ such that, for any $v_1, v_2, v_3 \in V_h$ and for $j = 1, \dots, N$,

$$\begin{aligned}
&\frac{1}{\Delta t}\left(u^{(n+1)} - u^{(n)}, v_1\right)_{I_j} = -\left(p^{(n+1/2)}, (v_1)_x\right)_{I_j} + \left[\hat{p}^{(n+1/2)} v_1\right]_{I_j}, \\
&\left(p^{(n+1/2)}, v_2\right)_{I_j} = \left(\frac{\partial H_{\mathrm{d}}}{\partial (u^{(n+1)}, u^{(n)})}, v_2 \right)_{I_j} + \left(\frac{\partial H_{\mathrm{d}}}{\partial (q^{(n+1)}, q^{(n)})}, (v_2)_x \right)_{I_j} - \left[\widehat{\frac{\partial H_{\mathrm{d}}}{\partial (q^{(n+1)}, q^{(n)})}} v_2 \right]_{I_j}, \\
&\left(q^{(n+1/2)}, v_3\right)_{I_j} = -\left(\left(\frac{u^{(n+1)} + u^{(n)}}{2} \right), (v_3)_x \right)_{I_j} + \left[\left(\frac{\hat{u}^{(n+1)} + \hat{u}^{(n)}}{2} \right) v_3 \right]_{I_j}.
\end{aligned}$$

The following energy-preservation property immediately holds.

Theorem 6.10. Assume that the fluxes are set to (6.52), (6.53), (6.54), and set such that (6.55), (6.56) hold. Then the solution of Scheme 4 satisfies

$$\frac{1}{\Delta t}\int_0^L \left(H(u^{(n+1)},q^{(n+1)})-H(u^{(n)},q^{(n)})\right)\mathrm{d}x = 0.$$

Dissipative cases

As was done for the conservative cases, we start the derivation of an energy-dissipative LDG scheme with an abstract semi-discrete scheme.

Semi-discrete scheme 5. Find $u(t,\cdot),p,q,r \in V_h$ such that, for any $v_1,v_2,v_3,v_4 \in V_h$ and for $j = 1,\dots,N$,

$$(u_t,v_1)_{I_j} = -(r,(v_1)_x)_{I_j} + [\hat{r}v_1]_{I_j}, \tag{6.58}$$

$$(r,v_2)_{I_j} = -(p,(v_2)_x)_{I_j} + [\hat{p}v_2]_{I_j}, \tag{6.59}$$

$$(p,v_3)_{I_j} = \left(\frac{\partial H}{\partial u},v_3\right)_{I_j} + \left(\frac{\partial H}{\partial q},(v_3)_x\right)_{I_j} - \left[\widehat{\frac{\partial H}{\partial q}}v_3\right]_{I_j}, \tag{6.60}$$

$$\left(q,v_4\right)_{I_j} = -\left(u,(v_4)_x\right)_{I_j} + \left[\hat{u}v_4\right]_{I_j}. \tag{6.61}$$

We then assume that the internal fluxes are given by, for $j = 1,\dots,N-1$,

$$\hat{r}_{j+1/2} = \eta r^+_{j+1/2} + (1-\eta)r^-_{j+1/2}, \tag{6.62}$$

$$\hat{p}_{j+1/2} = (1-\eta)p^+_{j+1/2} + \eta p^-_{j+1/2}, \tag{6.63}$$

$$\widehat{\frac{\partial H}{\partial q}}_{j+1/2} = \lambda\frac{\partial H}{\partial q}^{+}_{j+1/2} + (1-\lambda)\frac{\partial H}{\partial q}^{-}_{j+1/2}, \tag{6.64}$$

$$\hat{u}_{j+1/2} = (1-\lambda)u^+_{j+1/2} + \lambda u^-_{j+1/2}, \tag{6.65}$$

with real parameters η and λ, and boundary fluxes are set to satisfy

$$r^-_{N+1/2}p^-_{N+1/2} - \hat{r}_{N+1/2}p^-_{N+1/2} - r^-_{N+1/2}\hat{p}_{N+1/2} - r^+_{1/2}p^+_{1/2} + \hat{r}_{1/2}p^+_{1/2} + r^+_{1/2}\hat{p}_{1/2} = 0, \tag{6.66}$$

and again (6.56). Obviously (6.66) corresponds to $[rp]_0^L = 0$.

Theorem 6.11. Assume that the fluxes are set to (6.62), (6.63), (6.64), (6.65), and are set such that (6.66), (6.56) hold. Then the solution of Semi-discrete scheme 5 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x \le 0.$$

Proof. First, we note that for Semi-discrete scheme 5 satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x = -\|r\|^2_{L^2(0,L)} - \Theta_{rp} - \Theta_{uq},$$

where

$$\Theta_{rp} = \sum_{j=1}^N [rp - \hat{r}p - r\hat{p}]_{I_j}, \qquad \Theta_{uq} = \sum_{j=1}^N \left[u_t\frac{\partial H}{\partial q} - \hat{u}_t\frac{\partial H}{\partial q} - u_t\widehat{\frac{\partial H}{\partial q}}\right]_{I_j},$$

independently of the choice of fluxes. This can be checked as follows.

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L H(u,q)\,\mathrm{d}x \\
&=\sum_{j=1}^{N}\left\{\left(\frac{\partial H}{\partial u},u_t\right)_{I_j}+\left(\frac{\partial H}{\partial q},q_t\right)_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{\left(\frac{\partial H}{\partial u},u_t\right)_{I_j}-\left(\left(\frac{\partial H}{\partial q}\right)_x,u_t\right)_{I_j}+\left[\frac{\partial H}{\partial q}\hat{u}_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{(p,u_t)_{I_j}-\left(\frac{\partial H}{\partial q},u_{xt}\right)_{I_j}-\left(\left(\frac{\partial H}{\partial q}\right)_x,u_t\right)_{I_j}+\left[\frac{\partial H}{\partial q}\hat{u}_t+\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{-(r,p_x)_{I_j}+[\hat{r}p]_{I_j}-\left[\frac{\partial H}{\partial q}u_t-\frac{\partial H}{\partial q}\hat{u}_t-\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=\sum_{j=1}^{N}\left\{-(r,r)_{I_j}-[rp-\hat{r}p-r\hat{p}]_{I_j}-\left[\frac{\partial H}{\partial q}u_t-\frac{\partial H}{\partial q}\hat{u}_t-\widehat{\frac{\partial H}{\partial q}}u_t\right]_{I_j}\right\} \\
&=-\|r\|^2_{L^2(0,L)}-\Theta_{rp}-\Theta_{uq}.
\end{aligned}
$$

The first equality is a simple application of the chain rule. The second follows from (6.61) with $v_4 = \partial H/\partial q$. Substituting $v_3 = u_t$ in (6.60) leads to the third equality. Integrating the second term by parts and substituting $v_1 = p$ in (6.58), we obtain the fourth equality. Integrating the first term by-parts and substituting $v_2 = r$ in (6.59), we obtain the fourth equality.

Next, we show that $\Theta_{rp} = \Theta_{uq} = 0$. Since $\Theta_{uq} = 0$ under the assumptions (6.64), (6.65) and (6.56) was already proved previously, we here show only $\Theta_{rp} = 0$. Since

$$
\begin{aligned}
\Theta_{rp} &= \sum_{j=1}^{N}\left\{r^-_{j+1/2}p^-_{j+1/2}-\hat{r}_{j+1/2}p^-_{j+1/2}-r^-_{j+1/2}\hat{p}_{j+1/2}-r^+_{j-1/2}p^+_{j-1/2}+\hat{r}_{j-1/2}p^+_{j-1/2}+r^+_{j-1/2}\hat{p}_{j-1/2}\right\} \\
&= \sum_{j=1}^{N-1}\left\{\left(r^-_{j+1/2}p^-_{j+1/2}-r^+_{j+1/2}p^+_{j+1/2}\right)-\hat{r}_{j+1/2}\left(p^-_{j+1/2}-p^+_{j+1/2}\right)-\left(r^-_{j+1/2}-r^+_{j+1/2}\right)\hat{p}_{j+1/2}\right\} \\
&\quad + r^-_{N+1/2}p^-_{N+1/2}-\hat{r}_{N+1/2}p^-_{N+1/2}-r^-_{N+1/2}\widehat{p}_{N+1/2}-r^+_{1/2}p^+_{1/2}+\hat{r}_{1/2}p^+_{1/2}+r^+_{1/2}\widehat{p}_{1/2},
\end{aligned}
$$

$\Theta_{rp} = 0$ holds under the assumptions (6.62), (6.63) and (6.66). This completes the proof. □

Remark 6.14. Corresponding to Remark 6.13, and also in view of the Cahn–Hilliard example shown later, here we mention the choices of numerical fluxes when Neumann boundary conditions are imposed. In this case we set fluxes as follows: $\hat{r}_{1/2} = \hat{r}_{N+1/2} = 0$, $\hat{p}_{1/2} = p^+_{1/2}$, $\hat{p}_{N+1/2} = p^-_{N+1/2}$, $\hat{u}_{1/2} = u^+_{1/2}$, $\hat{u}_{N+1/2} = u^-_{N+1/2}$ and $\hat{q}_{1/2} = \hat{q}_{N+1/2} = 0$. Here we used the same, additional assumption as Remark 6.13 that $\partial H/\partial q \propto q$. It is obvious that the above choices satisfy the conditions (6.66) and (6.56).

We can now immediately derive an energy-dissipative fully-discrete scheme with the discrete partial derivatives.

Scheme 5. Find $u^{(n+1)}, p^{(n+1/2)}, q^{(n+1/2)}, r^{(n+1/2)} \in V_h$ such that, for any $v_1, v_2, v_3, v_4 \in V_h$ and for $j =$

$1,\dots,N$,

$$\begin{aligned}
&\frac{1}{\Delta t}\left(u^{(n+1)}-u^{(n)},v_1\right)_{I_j} = -\left(r^{(n+1/2)},(v_1)_x\right)_{I_j} + \left[\hat{r}^{(n+1/2)}v_1\right]_{I_j},\\
&\left(r^{(n+1/2)},v_2\right)_{I_j} = -\left(p^{(n+1/2)},(v_2)_x\right)_{I_j} + \left[\hat{p}^{(n+1/2)}v_2\right]_{I_j},\\
&\left(p^{(n+1/2)},v_3\right)_{I_j} = \left(\frac{\partial H_{\mathrm{d}}}{\partial(u^{(n+1)},u^{(n)})},v_3\right)_{I_j} + \left(\frac{\partial H_{\mathrm{d}}}{\partial(q^{(n+1)},q^{(n)})},(v_3)_x\right)_{I_j} - \left[\widehat{\frac{\partial H_{\mathrm{d}}}{\partial(q^{(n+1)},q^{(n)})}}v_3\right]_{I_j},\\
&\left(q^{(n+1/2)},v_4\right)_{I_j} = -\left(\left(\frac{u^{(n+1)}+u^{(n)}}{2}\right),(v_4)_x\right)_{I_j} + \left[\left(\frac{\hat{u}^{(n+1)}+\hat{u}^{(n)}}{2}\right)v_4\right]_{I_j}.
\end{aligned}$$

Theorem 6.12. Assume that the fluxes are set to (6.62), (6.63), (6.64), (6.65), and are set such that (6.66), (6.56) hold. Then the solution of Scheme 5 satisfies

$$\frac{1}{\Delta t}\int_0^L \left(H(u^{(n+1)},q^{(n+1)})-H(u^{(n)},q^{(n)})\right)\mathrm{d}x \le 0.$$

6.4.2 Applications to the KdV and Cahn–Hilliard equation

KdV equation

First, we consider the KdV equation under the periodic boundary conditions. The fully-discrete scheme is given in Scheme 4 with the energy-preserving fluxes (6.52), (6.53), (6.54) and the discrete partial derivatives

$$\begin{aligned}
\frac{\partial \mathcal{H}_{\mathrm{d}}}{\partial(u^{(n+1)},u^{(n)})} &= \left(u^{(n+1)}\right)^2 + u^{(n+1)}u^{(n)} + \left(u^{(n)}\right)^2,\\
\frac{\partial \mathcal{H}_{\mathrm{d}}}{\partial(q^{(n+1)},q^{(n)})} &= -\frac{q^{(n+1)}+q^{(n)}}{2}.
\end{aligned}$$

We check the qualitative behaviour of the numerical solutions. We consider the interaction of two solitons. The parameters were set to $x\in[0,20]$ ($L=20$), $\Delta x = 20/80$ ($N=80$), $\Delta t = 0.01$ and $\lambda = 0$. We set the initial value to $u(0,x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-4))+2\,\mathrm{sech}^2(x-12)$. The numerical solutions plotted in Figure 6.5 are qualitatively good. It is observed that the results become smooth as k increases, if we look them carefully.

We also check the convergence order in terms of spatial discretisations, by using P1, P2 and P3 elements. The initial value was set to $u(0,x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-5))+4\,\mathrm{sech}^2(\sqrt{2}(x+5))+4\,\mathrm{sech}^2(\sqrt{2}(x-15))$ with the spatial domain $x=[0,10]$. The last two terms were added so that the boundaries, i.e. $x=0,10$, are connected smoothly. We used the time stepsize $\Delta t = 10^{-4}$. Table 6.1 shows the results, where the order is calculated by

$$\text{order} = \frac{\log(\mathrm{err}(N)/\mathrm{err}(2N))}{\log 2}.$$

The expected order ($k+1$ for Pk elements) is observed.

The KdV equation has infinitely many conservation laws, such as

$$\mathcal{G} = \int_0^L \frac{u^2}{2}\,\mathrm{d}x.$$

For convenience, we call this quantity the norm. The Hamiltonian form associated with the norm is

$$u_t = \left((u\partial_x + \partial_x u) + \partial_x^3\right)\frac{\delta \mathcal{G}}{\delta u},$$

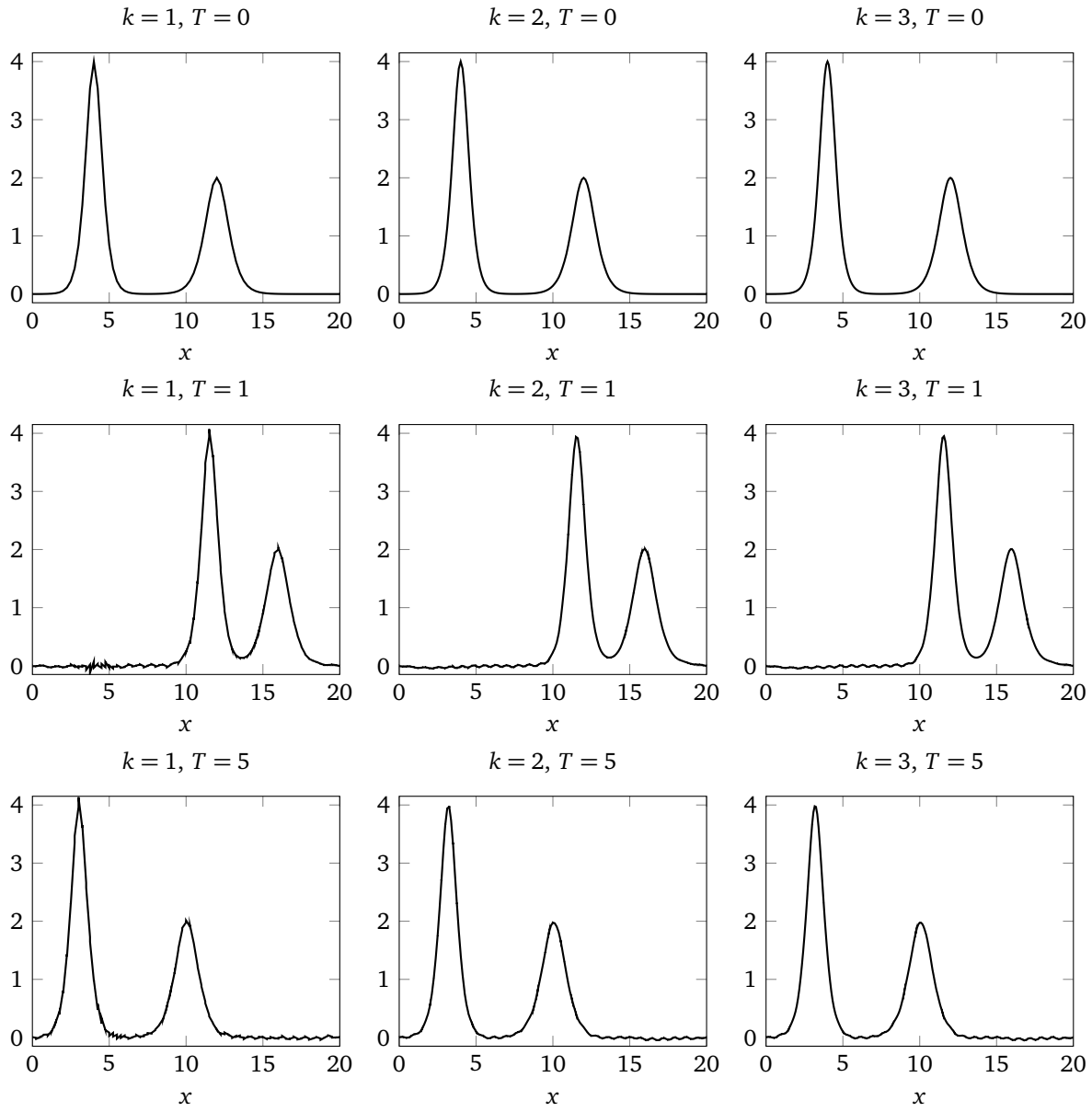


Figure 6.5: Numerical solutions at $T = 0, 1, 5$ obtained by the energy-preserving scheme ($k = 1, 2, 3$) for the KdV equation. The initial value was set to $u(0,x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-4)) + 2\,\mathrm{sech}^2(x-12)$. The parameters were set to $N = 80$, $\Delta t = 0.05$ and $\lambda = 0$.

Table 6.1: L^2-errors of the numerical solutions at the end time $T = 0.1$ by the energy-preserving scheme with polynomial degree $k = 1, 2, 3$, on uniform mesh. The initial values was set to $u(0,x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-5)) + 4\,\mathrm{sech}^2(\sqrt{2}(x+5)) + 4\,\mathrm{sech}^2(\sqrt{2}(x-15))$. Time stepsize was set to $\Delta t = 10^{-4}$.

	$k=1$		$k=2$		$k=3$	
N	error	order	error	order	error	order
10	4.5722	—	1.3489	—	6.8365e–2	—
20	4.7424e–1	3.2691	5.3649e–2	4.6521	4.7125e–3	3.8586
40	8.5016e–2	2.4798	4.4407e–3	3.5946	2.8043e–4	4.0707
80	1.9800e–2	2.1021	4.7646e–4	3.2203	1.7367e–5	4.0132

and a norm-preserving scheme can be derived based on this structure. The norm preserving scheme is given as follows: Find $u, r, q \in V_h$ such that, for any $v_1, v_2, v_3 \in V_h$ and for $j = 1, \dots, N$,

$$
\begin{aligned}
(u_t, v_1)_{I_j} &= -\big(3u^2, (v_1)_x\big)_{I_j} + \Big[\widehat{u^2} v_1\Big]_{I_j} - (r, (v_1)_x)_{I_j} + [\hat{r} v_1]_{I_j}, \\
(r, v_2)_{I_j} &= -(q, (v_2)_x)_{I_j} + [\hat{q} v_2]_{I_j}, \\
(q, v_3)_{I_j} &= -(u, (v_3)_x)_{I_j} + [\hat{u} v_3]_{I_j}.
\end{aligned}
$$

Note that $\widehat{u^2}$, a flux arising from the quadratic term u^2, is different from $(\hat{u})^2$. Norm-preserving fluxes are obtained to be

$$
\begin{aligned}
\widehat{u^2}_{j+1/2} &= \big(u^+_{j+1/2}\big)^2 + u^+_{j+1/2} u^-_{j+1/2} + \big(u^-_{j+1/2}\big)^2, \\
\hat{r}_{j+1/2} &= \frac{1}{2}\big(r^+_{j+1/2} + r^-_{j+1/2}\big), \\
\hat{q}_{j+1/2} &= \lambda q^+_{j+1/2} + (1-\lambda) q^-_{j+1/2}, \\
\hat{u}_{j+1/2} &= (1-\lambda) u^+_{j+1/2} + \lambda u^-_{j+1/2},
\end{aligned}
$$

with a real number λ. The midpoint rule for the temporal discretisation leads to a fully-discrete norm-preserving scheme.

Remark 6.15. Some norm-preserving DG schemes have been already proposed in [15, 214], but they are not LDG methods. It should also be mentioned that the norm-preserving H^1 weak form is not new (e.g., see [43, 65]), and was used to derive an LDG scheme in [212], but there the strict norm-preservation was not considered.

Cahn–Hilliard equation

Next, we consider the Cahn–Hilliard equation under the Neumann boundary conditions $u_x|_{x=0,L} = u_{xxx}|_{x=0,L} = 0$. The fully-discrete scheme is given in Scheme 5 with the discrete partial derivatives

$$
\begin{aligned}
\frac{\partial \mathcal{H}_\mathrm{d}}{\partial (u^{(n+1)}, u^{(n)})} &= \alpha \frac{u^{(n+1)} + u^{(n)}}{2} + \gamma \frac{\big(u^{(n+1)}\big)^3 + \big(u^{(n+1)}\big)^2 u^{(n)} + u^{(n+1)}\big(u^{(n)}\big)^2 + \big(u^{(n)}\big)^3}{4}, \\
\frac{\partial \mathcal{H}_\mathrm{d}}{\partial (q^{(n+1)}, q^{(n)})} &= -\beta \frac{q^{(n+1)} + q^{(n)}}{2}.
\end{aligned}
$$

The internal fluxes are set to (6.62), (6.63), (6.64), (6.65), and the boundary fluxes are set to $\hat{q}_{1/2} = \hat{q}_{N+1/2} = 0$, $\hat{r}_{1/2} = \hat{r}_{N+1/2} = 0$, $\hat{p}_{1/2} = p^+_{1/2}$, $\hat{p}_{N+1/2} = p^-_{N+1/2}$, $\hat{u}_{1/2} = u^+_{1/2}$, $\hat{u}_{N+1/2} = u^-_{N+1/2}$.

The computational parameters were set to $x \in [0, 1]$, $N = 40$, $\Delta t = 0.01$ and $\lambda = \eta = 0$. The initial value was set to

$$
u(0, x) = 0.1 \sin(2\pi x) + 0.01 \cos(4\pi x) + 0.06 \sin(4\pi x) + 0.02 \cos(10\pi x).
$$

with the parameters of the equation are $\alpha = -1$, $\beta = -0.001$, $\gamma = 1$. Numerical solutions are plotted in Figure 6.6, and the evolution of the energy is in Figure 6.7. The results are qualitatively good.

Remark 6.16. Although the fully-discrete scheme proposed by Xia et al. [202] is not an energy-dissipative scheme, the underlying semi-discrete scheme coincides with our semi-discrete scheme. This illustrates that the existing scheme can also be automatically derived based on our proposed method.

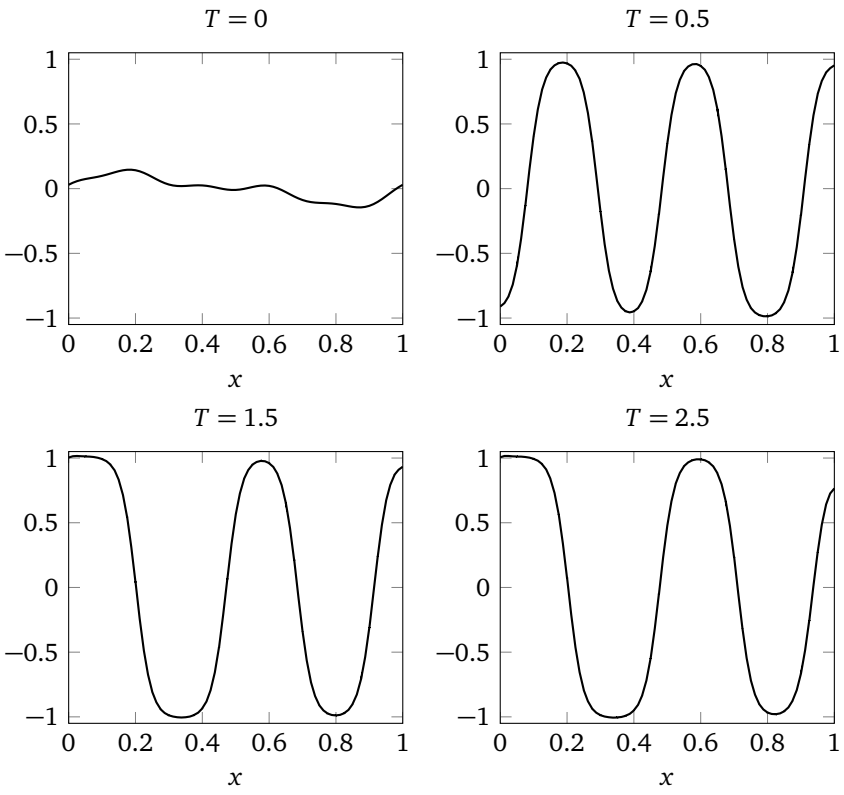


Figure 6.6: Numerical solutions obtained by the energy-dissipative scheme ($k = 2$) for the Cahn–Hilliard equation. The initial value was set to $u(0,x) = 0.1\sin(2\pi x) + 0.01\cos(4\pi x) + 0.06\sin(4\pi x) + 0.02\cos(10\pi x)$. The parameters were set to $N = 40$, $\Delta t = 0.01$ and $\lambda = \eta = 0$.

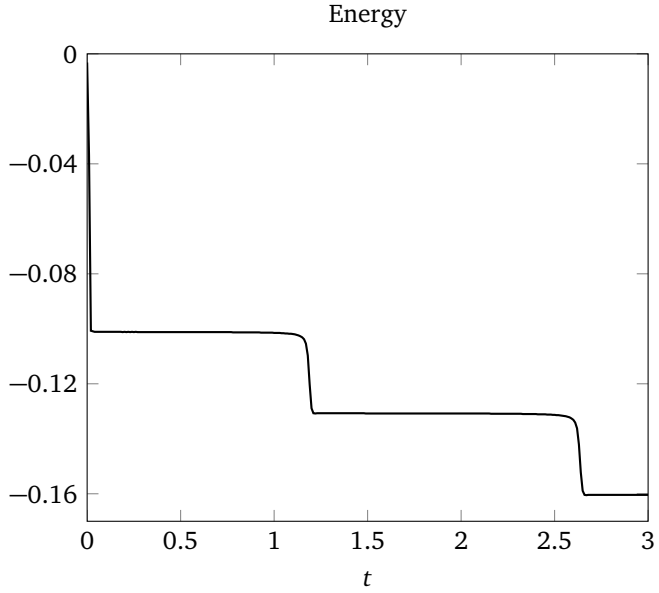


Figure 6.7: The evolution of the energy.

Chapter 7

Adaptivity in the Galerkin framework

In this chapter, an adaptive energy-preserving/dissipative discretisation method is proposed. The proposed method is a simple combination of existing moving grid methods and energy-preserving/dissipative method on nonuniform meshes. For simplicity, we consider only the Galerkin method.

In Section 7.1, we show a proposed adaptive numerical method. The method is discussed in detail in Sections 7.2 and 7.3. In Section 7.4, we present numerical experiments.

7.1 Adaptive energy-preserving/dissipative method

We propose an adaptive energy-preserving/dissipative numerical method. Firstly, we summarise the procedure of finite difference methods. The procedure consists of three steps.

1. Apply a suitable energy-preserving/dissipative method to (x_k^n, u_k^n) to obtain $(x_k^n, \tilde{u}_k^{n+1})$. Here, $u_k^n \approx u(t_n, x_k^n)$.
2. Use a suitable grid adaptation technique to obtain a new set of grid points $\{x_k^{n+1}\}$. Compute $(x_k^{n+1}, \hat{u}_k^{n+1})$ from $(x_k^n, \tilde{u}_k^{n+1})$ by a suitable interpolation, if $\hat{u}_k^{n+1}$ is used in the next step.
3. Compute (x_k^{n+1}, u_k^{n+1}) by projecting $(x_k^n, \tilde{u}_k^{n+1})$ or $(x_k^{n+1}, \hat{u}_k^{n+1})$ such that the energy of (x_k^{n+1}, u_k^{n+1}) is equal to that of $(x_k^n, \tilde{u}_k^n)$.

This can be extended to the finite element context in a straightforward way.

1. Apply a suitable energy-preserving/dissipative method to $u^{(n)}$ associated with the set of grid points $\{x_k^n\}$ to obtain $\tilde{u}^{(n+1)}$ associated with the grid $\{x_k^n\}$.
2. Use a suitable grid adaptation technique to obtain $\{x_k^{n+1}\}$. Compute $\hat{u}^{(n+1)}$ associated with $\{x_k^{n+1}\}$ by a suitable interpolation, if $\hat{u}^{(n+1)}$ is used in the next step.
3. Compute $u^{(n+1)}$ associated with $\{x_k^{n+1}\}$ by projecting $\tilde{u}^{(n+1)}$ or $\hat{u}^{(n+1)}$ such that the energy of $u^{(n+1)}$ is equal to that of $\tilde{u}^{(n)}$.

Obviously, the energy is preserved or dissipated in the first step, and is kept in the third step. Thus, the energy-preservation and dissipation properties are guaranteed for conservative and dissipative PDEs, respectively.

The new method differs from standard moving grid methods in the following two points. First, in the standard methods, the time-integrator is not restricted to be energy-preserving/dissipative. Second, the last step is usually not considered and $\hat{u}_k^{n+1}$ (or $\hat{u}^{(n+1)}$) is used as u_k^{n+1} (or $u^{(n+1)}$).

The proposed procedure has an obvious drawback that we have to solve nonlinear systems *twice* in each time step.

For conservative PDEs, this drawback can be weakened. In principle, we can use every (not always energy-preserving) time-integration method in the first step, by modifying the constrained condition of the last step: the energy of u_k^{n+1} (or $u^{(n+1)}$) is equal to that of previous time. However, because the first step takes a central role in the time integration, we suggest that one uses an energy-preserving method for qualitatively good numerical solution, especially when a large time stepsize is used. For dissipative PDEs, the use of a dissipative method in the first step is mandatory.

In the following two sections, we shall look at the procedures of the second and third steps in detail.

7.2 Moving grid methods

There have been a lot of research activities on moving grid methods. Among them, we survey two techniques. First, we summarise a method based on equidistribution. Next, we summarise a method based on wavelets. There are several differences between these approaches. In the first approach, the number of grid points is kept during the time evolution. Moreover, this approach can be easily incorporated with finite difference methods. In the second approach, the maximum of the total amount of grid points is fixed. This approach is for finite element methods. This section should be read as a survey.

7.2.1 Equidistribution

The following method is based on the equidistribution principle. The idea was first introduced by de Boor [16]. Set the domain to $[0, L]$. For a function $u(x)$ to be adapted and given function $\rho(x) > 0$, we divide the domain to N intervals $0 = x_0 < x_1 < \cdots < x_N = L$ such that

$$\int_{x_0}^{x_1} \rho(x)\,\mathrm{d}x = \int_{x_i}^{x_{i+1}} \rho(x)\,\mathrm{d}x, \qquad i = 1, \ldots, N-1.$$

The function $\rho(x)$ is called the monitor function or mesh density function (in the context of the latter case, $\rho^2(x)$ is called the monitor function). There are several typical choices of the monitor function, e.g.,

- Arc-length density function:

$$\rho(x) = \left(1 + u_x^2\right)^{\frac{1}{2}}, \tag{7.1}$$

- Curvature density function:

$$\rho(x) = \left(1 + u_{xx}^2\right)^{\frac{1}{4}}.$$

For a more detailed discussion including error estimates, see [110] for example.

7.2.2 Moving grid based on wavelets

A standard dynamic grid adaptation technique which is known in the context of wavelet based numerical methods [11, 191, 192, 193] is briefly reviewed without getting involved into the concept of wavelets.

Let $V^0 \subset V^1 \subset V^2 \subset \ldots$ which satisfy $\overline{\bigcup_{j=0}^{\infty} V^j} = L^2(0, L)$ be a sequence of finite dimensional function spaces, and $\{\phi_k^j(x)\}_k$ be the basis functions of V^j. The basis functions of W^j, the complement of V^j in V^{j+1}, i.e., $V^{j+1} = V^j \oplus W^j$, are denoted by $\{\psi_k^j(x)\}_k$.

Let J be a natural number. The function $u^J \in V^j$ which approximates $u \in L^2$ can be expressed as

$$u^J(x) = \sum_{k \in \mathcal{K}(J)} c_k^J \phi_k^J(x), \tag{7.2}$$

where $\mathcal{K}(J)$ denotes the set of indices of the basis functions. The definition (7.2) means that we use basic functions up to the depth J. Suppose that basis functions satisfy the interpolation property: $\phi_k^j(x_i^j) = \delta_{i,k}$ (x_i^j are the grid points). Then $u(x_k^J)$ can be chosen as the coefficient c_k^J. Since $V_J = V^0 \oplus W^0 \oplus \cdots \oplus W^{J-1}$, the approximate function (7.2) can also be rewritten as

$$u^J(x) = \sum_{k \in \mathcal{K}(0)} c_k^0 \phi_k^0(x) + \sum_{j=0}^{J-1} \sum_{k \in \mathcal{K}_C(j)} d_k^j \psi_k^j(x),$$

where $\mathcal{K}_C(j)$ denotes the set of indices of the basis functions$\{\psi_k^j\}_k$. The second term can further decomposed into a sum of two groups whose coefficients are above and below the threshold ϵ^j. This leads to

$$u^J(x) = \underbrace{\sum_{k \in \mathcal{K}(0)} c_k^0 \phi_k^0(x) + \sum_{j=0}^{J-1} \sum_{\substack{k \in \mathcal{K}_C(j) \\ |d_k^j| \geq \epsilon^j}} d_k^j \psi_k^j(x)}_{u_{\geq}^J(x)} + \underbrace{\sum_{j=0}^{J-1} \sum_{\substack{k \in \mathcal{K}_C(j) \\ |d_k^j| < \epsilon^j}} d_k^j \psi_k^j(x)}_{u_{<}^J(x)}. \tag{7.3}$$

When the threshold $\epsilon^j (\geq 0)$ is sufficiently small, $u_{\geq}^J(x)$ can be regarded as a rough approximation in V^J. Because of the interpolation property of the basis functions, each basis function corresponds to each grid point. Thus, extracting basis functions based on the decomposed form (7.3) implies the selection of the grid points, i.e., a grid adaptation (in terms of $u_{\geq}^J(x)$).

We usually assume that $\{\psi_k^j(x)\}_k$ also satisfy the interpolation property. But since explicit formulations can be quite cumbersome since they depend on $\{\phi_k^j(x)\}_k$ and the numbering of the indices, we illustrate this only by a concrete example in Remark 7.1.

Remark 7.1. In what follows, we assume as V^j the simplest one, i.e., the piecewise linear function space on uniform grids. Although it is rare to adopt P1 elements in the context of wavelet based numerical methods, this assumption makes the following discussion clear. In particular, we promise that V^j ($j = 0, 1, 2, \ldots$) is generated by recursively halving the subintervals. Let us, for example, V^0 consists of a single interval $[0, L]$, V^1 of two subintervals $[0, L/2]$ and $[L/2, L]$, and so on. (In the actual numerical experiment later on, we take finer grids as V^0 and consider its recursive divisions.) This automatically satisfies the assumption $V^0 \subset V^1 \subset V^2 \subset \cdots$. The basis functions are then given by

$$\phi_k^j(x) = \begin{cases} \dfrac{x - x_{k-1}^j}{x_k^j - x_{k-1}^j}, & x \in [x_{k-1}^j, x_k^j], \\ \dfrac{x_{k+1}^j - x}{x_{k+1}^j - x_k^j}, & x \in [x_k^j, x_{k+1}^j], \\ 0, & \text{otherwise}, \end{cases}$$

for $k = 0, \ldots, 2^j$ where $x_k^j = kL/2^j$ (note that special care for ϕ_0^j and $\phi_{2^j}^j$ is required to fit the boundary conditions, although at this point we do not get into its detail here). To help the readers' understanding, the basis functions of V^0, W^1, W^2 are illustrated in Figure 7.1. The coarsest space V^0 is spanned by ϕ_0^0 and ϕ_1^0, which correspond to the grid points x_0^0 and x_1^0, respectively. The first correction space W^0 has

a single new basis ψ_1^0. Note that $W^0 \subset V^1$ and ψ_1^0 is equivalent to ϕ_1^1. The grid points x_0^1 $(= x_0^0)$ and x_2^1 $(= x_1^0)$ are already employed in the coarser level V^0, and thus we do not consider them on this level (thus, accordingly, the corresponding bases ϕ_0^1 and ϕ_2^1 in V^1 are not taken into account as well; the grid names x_0^1 and x_2^1 are prepared just to make notation consistent). Similar observation can be done for W^2, where two new bases ψ_1^1 and ψ_2^1 (corresponding to the new grid points x_1^2 and x_3^2) are introduced. It is easy to examine that the interpolation property holds for them.

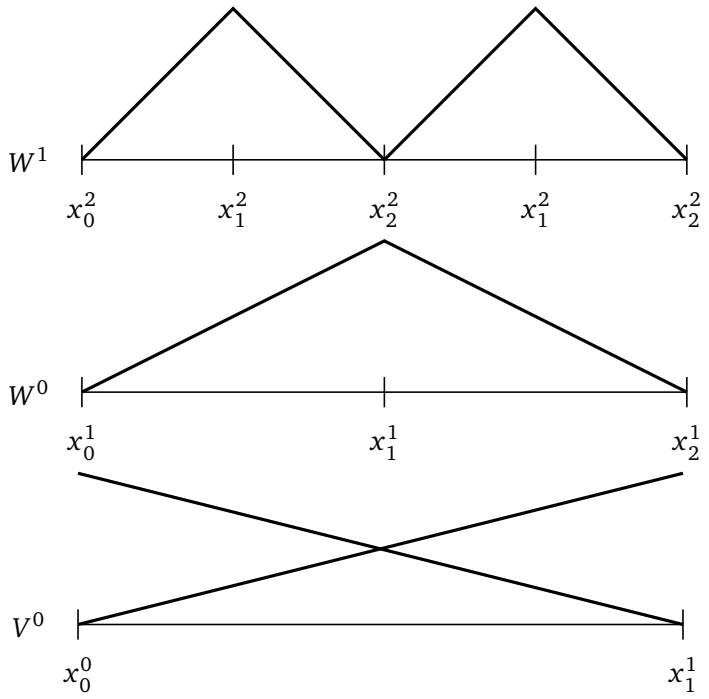


Figure 7.1: Bases of $[V^0]$ ϕ_0^0, ϕ_1^0; $[W^0]$ ψ_1^0 $(= \phi_1^1)$; $[W^1]$ ψ_1^1 $(= \phi_1^2)$, ψ_2^1 $(= \phi_3^2)$.

This example illustrates how the grid adaptation based on the decomposition (7.3) works. Suppose we work with the space V^J where the depth J is fixed. Then the following three are equivalent:

- dropping some set of basis functions $\{\psi_k^j(x)\}$ in V^J,
- dropping the corresponding grid points $\{x_k^j\}$,
- choosing a subspace of V^J comprised only of the remaining grid points.

In this sense, dropping the small terms in (7.3), i.e., $u_<^J(x)$, gives a grid adaptation based on the given function $u^J(x)$.

In more general wavelet context, the basis functions are usually derived from a so-called scaling function, see, e.g., [10, 192, 218].

Below we explain how we select appropriate grid points. Although this was already seen through (7.3), an additional technique is often incorporated for the quality of numerical solutions.

We introduce the concept of "adjacent zone." We say that the grid point x_i^s (or equivalently, the corresponding basis function ψ_i^s) belongs to the adjacent zone of x_k^j (or equivalently ψ_k^j), if the following relations are satisfied:

$$|s - j| \le M, \quad |x_i^s - x_k^j| \le a_j,$$

where a_j is the width of the adjacent zone, and M is the extent to which coarser and finer scales are taken into account.

In the wavelet based numerical methods, we first select the set of grid points based on (7.3), and then add the adjacent zone to the points. This strategy means, roughly speaking, the recovery of the dropped grid points (or corresponding bases) near the surviving points. Adding adjacent zone is not necessarily mandatory, but often makes the computation quite effective, since if the solution is moving to the left or right, it is better to include the grid points outside the surviving grid zone to catch such a behaviour accurately. The parameters M and a_j of adjacent zone should be carefully determined so that the algorithm catches the solution well in that sense. For example, it is good to set relatively large values for a problem whose solution drastically changes.

7.3 Projection

We explain the procedure of the last step of the proposed method. This step for finite element methods is formulated as a minimisation problem:

$$\begin{aligned} &\min \quad \left\| u^{(n+1)} - \tilde{u}^{(n+1)} \right\|, \\ &\text{s.t.} \quad \int_0^L H(u^{(n+1)})\,\mathrm{d}x = \int_0^L H(\tilde{u}^{(n+1)})\,\mathrm{d}x, \end{aligned}$$

or

$$\begin{aligned} &\min \quad \left\| u^{(n+1)} - \hat{u}^{(n+1)} \right\|, \\ &\text{s.t.} \quad \int_0^L H(u^{(n+1)})\,\mathrm{d}x = \int_0^L H(\tilde{u}^{(n+1)})\,\mathrm{d}x. \end{aligned}$$

A similar formulation for finite difference methods is straightforward. The simplest way of solving the above problem is to apply a existing solver for constrained nonlinear equations such as fmincon of matlab. The minimisation problem can also be solved by the Lagrangian multiplier technique, whose computation is faster than the first approach in most cases. However, special care must be taken for both approaches. Our preliminary experiments demonstrated that iterations of both approaches sometimes failed to converge or decrease the residual sufficiently. These drawbacks were sometimes avoided successfully by changing the initial value of the iteration, or using a relatively large tolerance, say 10^{-6}.

7.4 Numerical experiments

We show numerical experiments for the KdV and Cahn–Hilliard equations.

KdV equation

For the KdV equation

$$u_t = \partial_x\big(3u^2 + u_{xx}\big),$$

with the periodic boundary condition, we show the numerical results by the equidistribution approach.

Note that we can intuitively understand that the wavelet based moving grid method is less effective. We are forced to use a very small time stepsize or chose wide adjacent zones in order to capture a soliton because the speed of the wave could be very fast. Thus, we consider the equidistribution approach. As mentioned in Remark 5.2, the use of the equidistribution approach to the KdV equation was originally due to [71]. Results by the wavelet based moving grid method are found in [152].

We set the domain to $[0, 10]$ and consider the initial value to $u(0, x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-4))$. The number of grid points is 100. The arc-length density function (7.1) is chosen as a mesh density function.

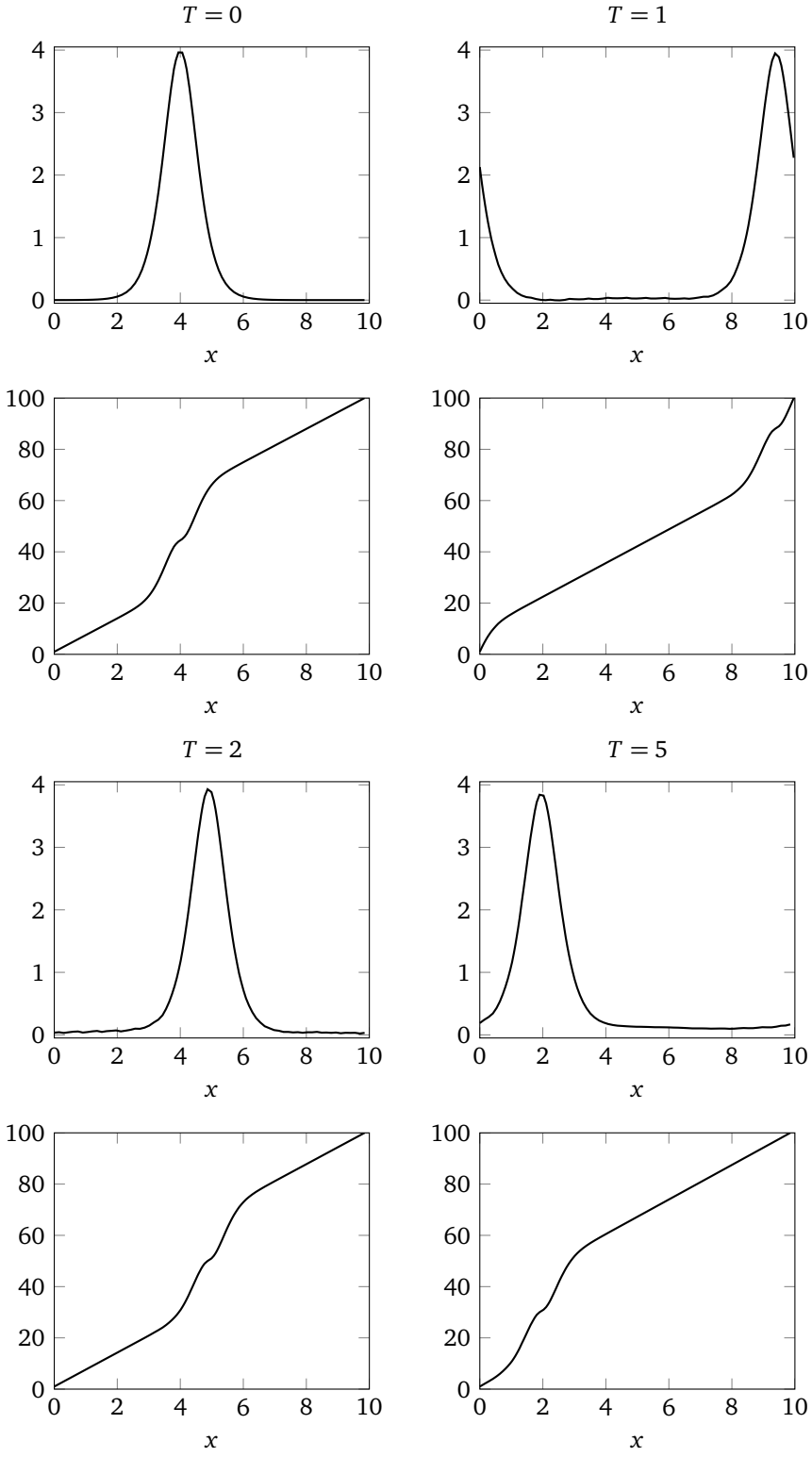


Figure 7.2: Evolution of the numerical solution for 1-soliton of the KdV equation. Each figure below a numerical solution shows the number of cumulative grid points. The grid points are dense where the slope is steep. The time stepsize was set to $\Delta t = 0.02$.

Results are plotted in Figure 7.2. There, the second figure of each time shows the number of cumulative grid points. Hence, the grid points are dense where the slope is steep. It is observed that the grid points are dense around the soliton at each time. Note that the slope is less steep at the centre of the soliton.

Similar results are also observed for 2-soliton solutions as shown in Figure 7.3, where the domain, initial value and the number of grid points were set to $[0, 20]$, $u(0, x) = 4\,\mathrm{sech}^2(\sqrt{2}(x-4)) + 2\,\mathrm{sech}^2(x-12)$ and 200, respectively.

Cahn–Hilliard equation

For the Cahn–Hilliard equation

$$u_t = \partial_x^2\big(\alpha u + \gamma u^3 + \beta u_{xx}\big),$$

with the boundary condition $u_x|_{x=0,L} = u_{xxx}|_{x=0,L} = 0$, we show the numerical results by the wavelet-based approach. The initial value was set to $u(0, x) = \cos(2\pi x)$ in the domain $[0, 1]$. Results for different thresholds are plotted in Figure 7.4 and Figure 7.5. More grid points are used when the threshold is small. It is also observed, for both cases, the grid points are moved to an appropriate area: at $T = 0.5$ the grid points are dense where the slope of u is sharp.

Discussions

Although we saw some nice snapshots in the above examples, it is too early to say that the proposed algorithm is more practical than the standard energy-preserving/dissipative methods. The main reason is that we have to solve nonlinear systems twice per each time step, and the computation of the projection is heavier than that of the time stepping part. As a future work, the algorithms for solving the projection part should be investigated.

Assume that we have an algorithm solving the projection part, whose computational cost is comparable with the simplified Newton method for the time stepping part. Also assume that their costs are $\mathcal{O}(N^3)$, where N is the size of the nonlinear system (i.e., the number of grid points). Then it is roughly estimated that the overall computation of the proposed method is faster than the standard methods, if the number of grid points can be constantly reduced to $(1/2)^{(1/3)} \approx 0.7973$ times the number at the initial time.

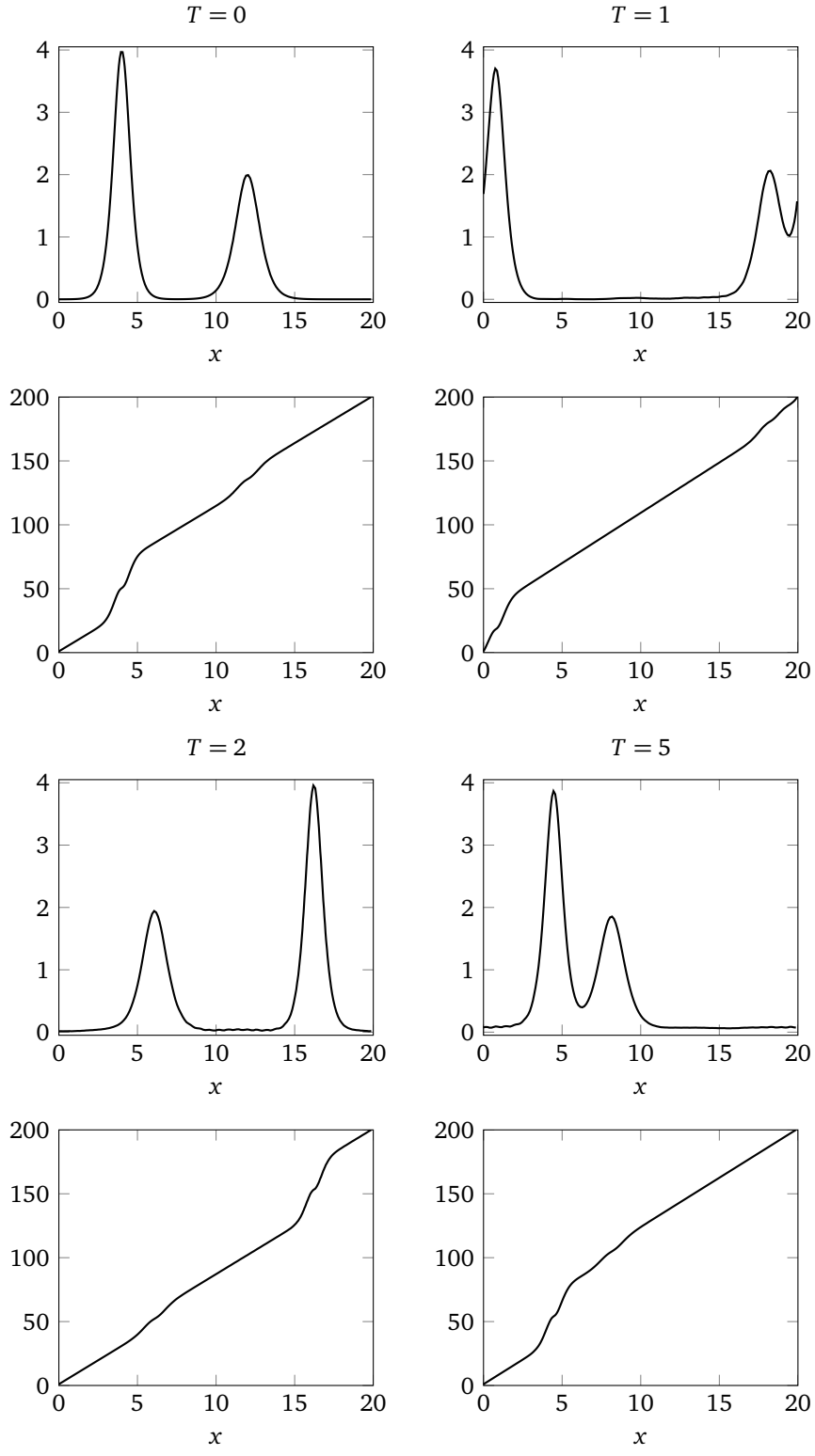


Figure 7.3: Evolution of the numerical solution for 2-soliton of the KdV equation. Each figure below a numerical solution shows the number of cumulative grid points. The grid points are dense where the slope is steep. The time stepsize was set to $\Delta t = 0.02$.

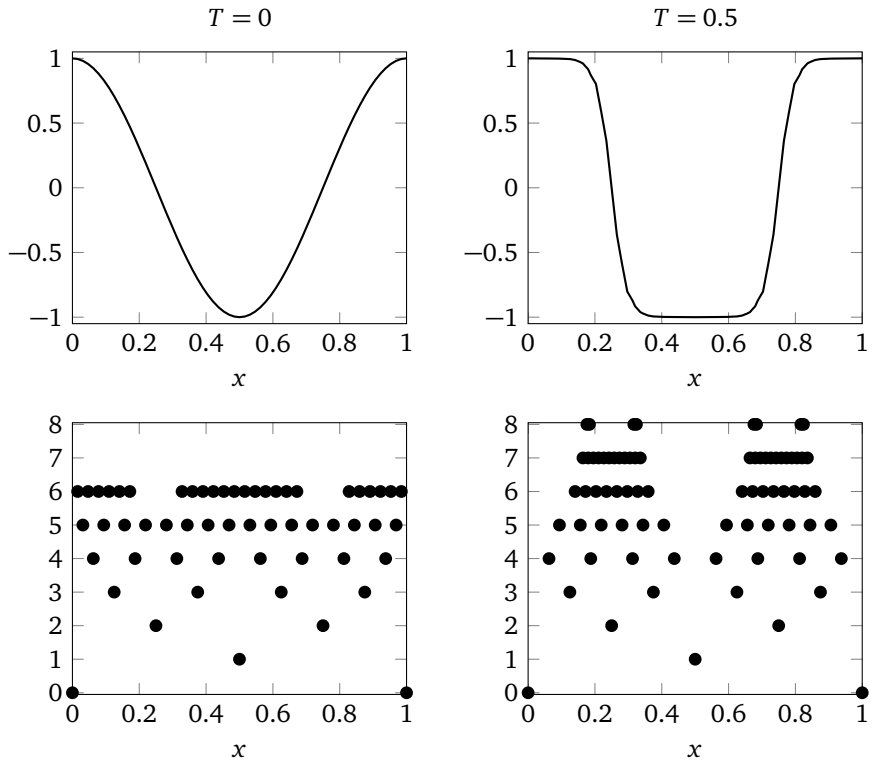


Figure 7.4: Evolution of the numerical solution and the set of grid points for the Cahn–Hilliard equation. The wavelet-based grid adaptation technique was used. The parameters were set to $\Delta t = 0.01$, $J = 8$, $\epsilon^j = 10^{-2}$, $M = 1$ and $a_j = 1/2^j$.

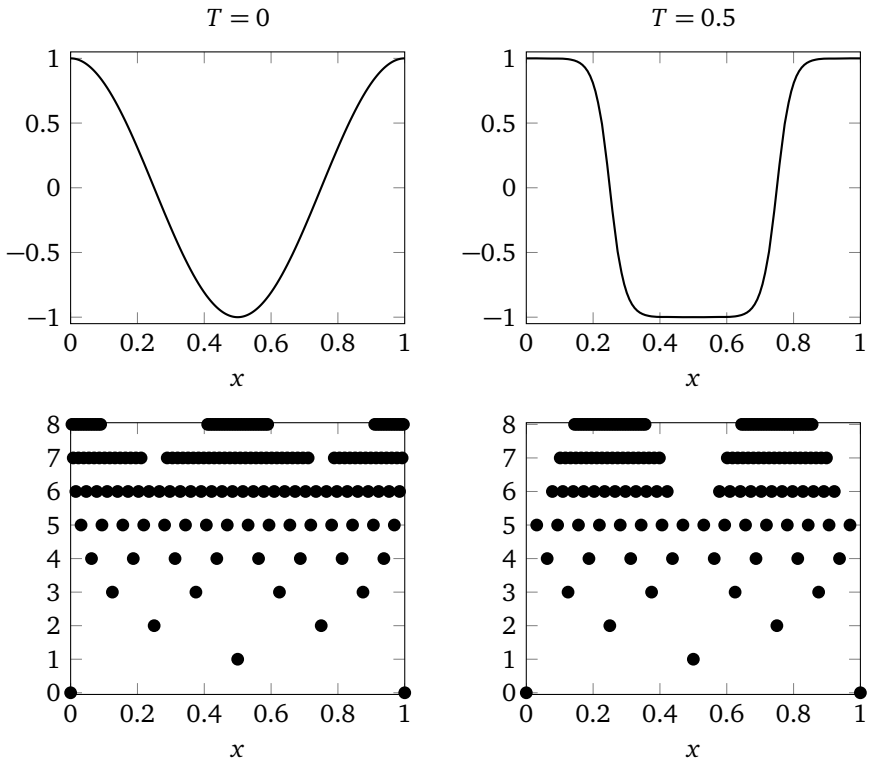


Figure 7.5: Evolution of the numerical solution and the set of grid points for the Cahn–Hilliard equation. The wavelet-based grid adaptation technique was used. The parameters were set to $\Delta t = 0.01$, $J = 8$, $\epsilon^j = 10^{-3}$, $M = 1$ and $a_j = 1/2^j$.

Chapter 8

Geometric integrators for Hunter–Saxton like equations

Contents of Chapter 8 are not publicised, because this chapter is a work of joint authorship and the publication is not approved by a co-author.

Chapter 9

Conclusion and future prospects

In this thesis, we have developed several geometric numerical integration methods for energy-driven evolution equations. We briefly summarise the main results of each chapter and discuss future prospects.

Part I

Chapter 3 The main motivation of Chapter 3 was to construct efficient energy-preserving numerical methods for Hamiltonian systems with periodic/oscillatory solutions. For this aim, we first gave in Theorem 3.2 a new characterisation of continuous stage Runge–Kutta methods being energy-preserving for Hamiltonian systems. By combining the characterisation with the idea of exponential-fitted methods, we succeeded in designing energy-preserving exponentially-fitted numerical methods. We then extended the new methods to Poisson systems. As our working example, we derived second- and fourth-order integrators. As our future work, we should study a more systematic strategy for deriving higher-order integrators. In addition, we showed that trigonometric energy-preserving integrators can be constructed for oscillatory Hamiltonian systems.

Chapter 4 By using the energy-preserving condition Theorem 3.2 and characterisation of the order conditions in terms of the coefficient polynomial of CSRK methods, we constructed a new class of energy-preserving integrators for Hamiltonian systems. The main advantage of the new method is that it is much faster than the average vector field collocation (i.e., existing energy-preserving) method (Definition 2.10). The computational cost of the new method becomes further reduced when parallelism is available. As our working example, we derived fourth- and sixth-order integrators. Our future works include the following.

- Although we newly derived up to sixth-order integrators, it seems quite complicated to obtain higher-order integrators because the characteristic polynomial appearing in the derivation becomes cumbersome. It would be of interest to consider a systematic strategy for deriving intended high order integrators.
- The advantage of the new method becomes notable for large scale problems. Thus, the method should be applied to such problems, e.g., semi-discrete schemes for two- or three-dimensional Hamiltonian PDEs, in order to estimate practical efficiency.

Part II

Chapter 6 We constructed a general framework for deriving energy-preserving/dissipative H^1 Galerkin schemes. The new framework can be applicable to PDEs with complicated variational structures. We further combined the framework with the idea of discontinuous Galerkin methods, which allows us to implement spatially high-order energy-preserving/dissipative schemes easily.

This chapter ignored mathematical analyses such as unique solvability, existence and convergence. In particular, backward error analysis for both spatial and temporal discretisations would be interesting. For these aims, properties of underlying weak-forms should also be investigated.

Chapter 7 We combined energy conservative/dissipative numerical methods on static nonuniform grids with grid adaptation techniques. We applied the proposed method to the KdV and Cahn–Hilliard equations. As a future work, convergence analysis would also be a challenging topic. One can apply the standard finite element theory to the time stepping part. On the other hand, analysis of the optimisation part would be difficult and quite challenging.

Chapter 8 We constructed several structure-preserving integrators for Hunter–Saxton like equations. It was not obvious how we can handle a nonlocal operator ∂_x^{-2} in the context of geometric numerical integration methods. This problem was successfully settled for some boundary conditions and target geometric properties. It would be of interest to apply the presented techniques to other PDEs with the same nonlocal operator ∂_x^{-2} and get more insight into the behaviour of such equations. Furthermore, the treatment of other nonlocal operators in the context of geometric numerical integration methods should also be studied.

Finally, we discuss future perspectives from a broader point of view.

The main purpose of geometric numerical integration methods is to simulate practical evolution equations efficiently over a long period of time, and analyse the methods mathematically. It is strongly hoped that some existing methods (including new methods presented in this thesis) will be extended in various ways so that they solve large-scale problems efficiently. Most of the contents of this thesis aimed to contribute to this purpose in a relatively fundamental level, and these studies should be further pushed forward from a practical, as well as theoretical, point of view.

In the context of energy-preserving temporal discretisation, relaxation methods would be an interesting topic. As mentioned in Remark 2.5, second order linearly implicit methods have been considered by relaxing the exact energy-preservation. It is hoped that high order methods can also be constructed, however, no one has succeeded in deriving such methods. Numerical methods preserving multiple first integrals would also be an interesting topic. There are a number of studied on this topic. However, all of the existing methods have the drawback that they are not B-series methods, and thus it is difficult to study the long time behaviour. Therefore, we have to study if it is possible to construct B-series integrators preserving multiple first integrals. If it is impossible, a new analysis method for existing methods will be hoped.

In the context of spatial discretisation, we should further develop efficient discretisation techniques and analyse them.

In this thesis, we dealt with only deterministic problems. However, there are a number of non-deterministic problems such as stochastic differential equations. Moreover, practical problems do not always possess nice geometric properties. Therefore, it would be interesting to consider to what extent the idea of geometric numerical integration methods is useful for such problems. Indeed, geometric numerical integration methods for stochastic equations have recently been attracting attention, and similar studies should be carried out for other type of equations.

Bibliography

[1] Y. Aimoto, T. Matsuo and Y. Miyatake: A local discontinuous Galerkin method based on variational structure, to appear in Discrete Contin. Dyn. Syst. Ser. S. (see p. 8).

[2] G. D. Akrivis, V. A. Dougalis and O. A. Karakashian: On fully discrete Galerkin methods of second-order temporal accuracy for the nonlinear Schrödinger equation, *Numer. Math.* **59** (1991) 31–53 (see p. 7).

[3] V. I. Arnold: *Mathematical Methods of Classical Mechanics*. 2nd ed. Springer-Verlag, New York, 1989 (see p. 23).

[4] U. M. Ascher, H. Chin and S. Reich: Stabilization of DAEs and invariant manifolds, *Numer. Math.* **67** (1994) 131–149 (see p. 34).

[5] U. M. Ascher and R. I. McLachlan: Multisymplectic box schemes and the Korteweg–de Vries equation, *Appl. Numer. Math.* **48** (2004) 255–269 (see p. 73).

[6] U. M. Ascher and S. Reich: On some difficulties in integrating highly oscillatory Hamiltonian systems, in *Computational molecular dynamics: challenges, methods, ideas (Berlin, 1997)*. **4**. Lect. Notes Comput. Sci. Eng. Springer-Verlag, Berlin, 1999, 281–296 (see p. 34).

[7] K. Atkinson and W. Han: *Theoretical Numerical Analysis: A Functional Analysis Framework*. 3rd ed. Springer-Verlag, Heidelberg, 2009 (see p. 96).

[8] F. Bassi and S. Rebay: A high-orderd accurate discontinuous finite element method for the numerical olution of the compressible Navier–Stokes equations, *J. Comput. Phys.* **131** (1997) 267–279 (see p. 110).

[9] J. Baumgarte: Stabilization of constraints and integrals of motion in dynamical systems, *Comput. Methods in Appl. Mech. Eng.* **1** (1972) 1–16 (see p. 34).

[10] S. Bertoluzza and G. Naldi: A wavelet collocation method for the numerical solution of partial differential equations, *Appl. Comput. Harmon. Anal.* **3** (1996) 1–9 (see p. 126).

[11] G. Beylkin and J. M. Keiser: On the adaptive numerical solution of nonlinear partial differential equations in wavelet bases, *J. Comput. Phys.* **132** (1997) 233–259 (see p. 124).

[12] S. Blanes, F. Casas, P. Chartier and A. Murua: Optimized high-order splitting methods for some classes of parabolic equations, *Math. Comput.* **82** (2013) 1559–1576 (see p. 21).

[13] S. Blanes, F. Casas and A. Murua: Splitting methods with complex coefficients, *Bol. Soc. Esp. Mat. Apl.* **50** (2010) 47–60 (see p. 21).

[14] P. B. Bochev and C. Scovel: On quadratic invariants and symplectic structure, *BIT* **34** (1994) 337–345 (see p. 31).

[15] J. L. Bona, H. Chen, O. Karakashian and Y. Xing: Conservative, discontinuous Galerkin-methods for the generalized Korteweg–de Vries equation, *Math. Comput.* **82** (2013) 1401–1432 (see pp. 110, 121).

[16] C. de Boor: Good approximation by splines with variable knots, in *Spline Functions and Approximation Theory*. **21**. ISNM International Series of Numerical Mathematics. Birkhäuser Basel, 1973, 57–72 (see p. 124).

[17] A. Bressan and A. Constantin: Global solutions of the Hunter–Saxton equation, *SIAM J. Math. Anal.* **37** (2005) 996–1026 (see p. 85).

[18] T. J. Bridges: A geometric formulation of the conservation of wave action and its implications for signature and the classification of instabilities, *Proc. Roy. Soc. London Ser. A* **453** (1997) 1365–1395 (see pp. 7, 72, 73).

[19] T. J. Bridges and S. Reich: Multi-symplectic integrators: numerical schemes for Hamiltonian PDEs that conserve symplecticity, *Phys. Lett. A* **284** (2001) 184–193 (see pp. 7, 72, 73, 80).

[20] T. J. Bridges and S. Reich: Numerical methods for Hamiltonian PDEs, *J. Phys. A* **39** (2006) 5287 (see p. 74).

[21] L. Brugnano and F. Iavernaro: Line integral methods which preserve all invariants of conservative problems, *J. Comput. Appl. Math.* **236** (2012) 3905–3919 (see p. 34).

[22] L. Brugnano, F. Iavernaro and D. Trigiante: Hamiltonian boundary value methods (energy preserving discrete line integral methods), *J. Numer. Anal. Ind. Appl. Math.* **5** (2010) 17–37 (see p. 37).

[23] L. Brugnano, F. Iavernaro and D. Trigiante: A note on the efficient implementation of Hamiltonian BVMs, *J. Comput. Appl. Math.* **236** (2011) 375–383 (see p. 37).

[24] L. Brugnano and Y. Sun: Multiple invariants conserving Runge–Kutta type methods for Hamiltonian problems, *Numer. Algor.* **65** (2014) 611–632 (see p. 34).

[25] K. Burrage and J. Butcher: Stability criteria for implicit Runge–Kutta methods, *SIAM J. Numer. Anal.* **16** (1979) 46–57 (see p. 32).

[26] J. C. Butcher: Implicit Runge–Kutta processes, *Math. Comput.* **18** (1964) 50–64 (see pp. 1, 18).

[27] J. C. Butcher: A modified multistep method for the numerical integration of ordinary differential equations, *J. ACM* (1965) (see p. 14).

[28] J. C. Butcher: The effective order of Runge–Kutta methods, in *Conference on the Numerical Solution of Differential Equations*. **109**. Lecture Notes in Mathematics. Springer-Verlag, Heidelberg, 1969, 133–139 (see p. 21).

[29] J. C. Butcher: An algebraic theory of integration methods, *Math. Comput.* **26** (1972) 79–106 (see p. 37).

[30] J. C. Butcher: A stability property of implicit Runge–Kutta methods, *BIT* **15** (1975) 358–361 (see p. 32).

[31] J. C. Butcher: General linear methods, *Acta Numerica* **15** (2006) 157–256 (see p. 14).

[32] J. C. Butcher: *Numerical Methods for Ordinary Differential Equations*. 2nd ed. Wiley, Chichester, 2008 (see pp. 1, 14, 32).

[33] J. C. Butcher and G. Imran: Symplectic effective order methods, *Numer. Algor.* **65** (2014) 499–517 (see p. 41).

[34] J. C. Butcher and Y. Miyatake: Novel energy-preserving integrators for Hamiltonian systems, in preparation (see p. 8).

[35] G. D. Byrne and R. J. Lambert: Pseudo-Runge–Kutta methods involving two points, *J. ACM* **13** (1966) 114–123 (see p. 14).

[36] M. Calvo, J. M. Franco, J. I. Montijano and L. Rández: Sixth-order symmetric and symplectic exponentially fitted modified Runge–Kutta methods of Gauss type, *Comput. Phys. Comm.* **178** (2008) 732–744 (see pp. 40, 44, 46).

[37] M. Calvo, J. M. Franco, J. I. Montijano and L. Rández: Structure preservation of exponentially fitted Runge–Kutta methods, *J. Comput. Appl. Math.* **218** (2008) 421–434 (see pp. 40, 46).

[38] M. Calvo, J. M. Franco, J. I. Montijano and L. Rández: Sixth-order symmetric and symplectic exponentially fitted Runge–Kutta methods of the Gauss type, *J. Comput. Appl. Math.* **178** (2009) 387–398 (see pp. 40, 46, 58, 59).

[39] M. Calvo, J. M. Franco, J. I. Montijano and L. Rández: On high order symmetric and symplectic trigonometrically fitted Runge–Kutta methods with an even number of stages, *BIT* **50** (2010) 3–21 (see pp. 40, 46).

[40] M. Calvo, J. M. Franco, J. I. Montijano and L. Rández: Symmetric and symplectic exponentially fitted Runge–Kutta methods of high order, *Comput. Phys. Comm.* **181** (2010) 2044–2056 (see pp. 40, 45, 46).

[41] R. Camassa and D. D. Holm: An integrable shallow water equation with peaked solitons, *Phys. Rev. Lett.* **71** (1993) 1661–1664 (see p. 83).

[42] R. Camassa, D. D. Holm and J. M. Hyman: A new integrable shallow water equation, *Adv. Appl. Mech.* **31** (1994) 1–33 (see p. 83).

[43] G. F. Carey and Y. Shen: Approximations of the KdV equation by least squares finite elements, *Comput. Methods in Appl. Mech. Eng.* **93** (1991) 1–11 (see p. 121).

[44] A. Cayley: XXVIII. On the theory of the analytical forms called trees, *Philosophical Magazine Series 4* **13** (1857) 172–176 (see p. 17).

[45] E. Celledoni, V. Grimm, R. I. McLachlan, D. I. McLaren, D. O'Neale, B. Owren and G. R. W. Quispel: Preserving energy resp. dissipation in numerical PDEs using the "Average Vector Field" method, *J. Comput. Phys.* **231** (2012) 6770–6789 (see p. 76).

[46] E. Celledoni, H. Marthinsen and B. Owren: An introduction to Lie group integrators – basics, new developments and applications, *J. Comput. Phys.* **257** (2014) 1040–1061 (see p. 34).

[47] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, G. R. W. Quispel and W. M. Wright: Energy-preserving Runge–Kutta methods, *ESIAM Math. Model. Numer. Anal.* **43** (2009) 645–649 (see p. 34).

[48] E. Celledoni, B. Owren and Y. Sun: The minimal stage, energy preserving Runge–Kutta method for polynomial Hamiltonian systems is the averaged vector field method, *Math. Comput.* **83** (2014) 1689–1700 (see p. 35).

[49] P. Chartier, E. Faou and A. Murua: An algebraic approach to invariant preserving integators: the case of quadratic and Hamiltonian invariants, *Numer. Math.* **103** (2006) 575–590 (see p. 4).

[50] Y. Chung, C. K. R. T. Jones, T. Schäfer and C. E. Wayne: Ultra-short pulses in linear and nonlinear media, *Nonlinearity* **18** (2005) 1351–1374 (see p. 84).

[51] B. Cockburn, G. E. Karniadakis and C.-W. Shu: *Discontinuous Galerkin Methods, Theory, Computation and Applications*. Springer-Verlag, Heidelberg, 2000 (see p. 110).

[52] B. Cockburn and C.-W. Shu: The local discontinuous Galerkin method for time-dependent convection-diffusion systems, *SIAM J. Numer. Anal.* **35** (1998) 2440–2463 (see p. 110).

[53] D. Cohen and E. Hairer: Linear energy-preserving integrators for Poisson systems, *BIT* **51** (2011) 91–101 (see pp. 39, 40).

[54] D. Cohen, T. Jahnke, K. Lorenz and C. Lubich: Numerical integrators for highly oscillatory Hamiltonian systems: a review, in *Analysis, modeling and simulation of multiscale problems*. Springer-Verlag, Berlin, 2006, 553–576 (see p. 40).

[55] D. Cohen, T. Matsuo and X. Raynaud: A multi-symplectic numerical integrator for the two-component Camassa–Holm equation, *J. Nonlinear Math. Phys.* **21** (2014) 442–453 (see p. 82).

[56] D. Cohen, B. Owren and X. Raynaud: Multi-symplectic integration of the Camassa–Holm equation, *J. Comput. Phys.* **227** (2008) 5492–5512 (see pp. 82, 85).

[57] D. Cohen and X. Raynaud: Geometric finite difference schemes for the generalized hyperelastic-rod wave equation, *J. Comput. Appl. Math.* **235** (2011) 1925–1940 (see p. 78).

[58] A. Constantin and J. Escher: Global existence and blow-up for a shallow water equation, *Ann. Scuola Norm. Sup. Pisa Cl. Sci.* **26** (1998) 303–328 (see p. 105).

[59] A. Constantin and W. A. Strauss: Stability of peakons, *Comm. Pure Appl. Math.* **53** (2000) 603–610 (see p. 105).

[60] M. Crouzeix: Sur la B-stabilité des méthodes de Runge–Kutta, *Numer. Math.* **32** (1979) 75–82 (see p. 32).

[61] M. Dahlby and B. Owren: A general framework for deriving integral preserving numerical methods for PDEs, *SIAM J. Sci. Comput.* **33** (2011) 2318–2340 (see pp. 35, 78).

[62] M. Dahlby, B. Owren and T. Yaguchi: Preserving multiple first integrals by discrete gradients, *J. Phys. A* **4** (2011) 305205 (see p. 34).

[63] G. Dahlquist: A special stability problem for linear multistep methods, *BIT* **3** (1963) 27–43 (see p. 14).

[64] G. G. Dahlquist: Stability and error bounds in the numerical integration of ordinary differential equations, PhD Thesis, University of Stockholm, 1958 (see p. 1).

[65] A. Debussche and J. Printems: Numerical simulation of the stochastic Korteweg–de Vries equation, *Phys. D* **134** (1999) 200–226 (see p. 121).

[66] A. Degasperis and M. Procesi: Asymptotic integrability, in *Symmetry and perturbation theory (Rome, 1998)*. World Sci. Publ., River Edge, NJ, 1999, 23–37 (see p. 83).

[67] M. Delfour, M. Fortin and G. Payr: Finite-difference solutions of a non-linear Schrödinger equation, *J. Comput. Phys.* **44** (1981) 277–288 (see p. 7).

[68] P. Deuflhard: A study of extrapolation methods based on multistep schemes without parasitic solutions, *Z. Angew. Math. Phys.* **30** (1979) 177–189 (see pp. 40, 61).

[69] Q. Du and R. A. Nicolaides: Numerical analysis of a continuum model of phase transition, *SIAM J. Numer. Anal.* **28** (1991) 1310–1322 (see pp. 7, 82).

[70] E. Eich: Convergence results for a coordinate projection method applied to mechanical systems with algebraic constraints, *SIAM J. Numer. Anal.* **30** (1993) 1467–1482 (see p. 34).

[71] S. Eidnes: Integral preserving numerical methods on moving grids, Master Thesis, Norwegian University of Science and Technology, 2013 (see pp. 84, 127).

[72] K. Feng: On difference schemes and symplectic geometry, in *Proceedings of the 1984 Beijing symposium on differential geometry and differential equations*. Science Press, Beijing, 1985, 42–58 (see p. 1).

[73] K. Feng: Difference schemes for Hamiltonian formalism and symplectic geometry, *J. Comput. Math.* **4** (1986) 279–289 (see p. 1).

[74] K. Feng and M. Qin: *Symplectic Geometric Algorithms for Hamiltonian Systems*. Springer-Verlag, Heidelberg, 2010 (see p. 69).

[75] E. Fermi, J. Pasta and S. Ulam: Studies of nonlinear problems, tech. rep. Los Alamos Scientific Laboratory Report No. LA-1940, 1955 (see p. 25).

[76] J. M. Franco: An embedded pair of exponentially fitted explicit Runge–Kutta methods, *J. Comput. Appl. Math.* **149** (2002) 407–414 (see pp. 40, 44).

[77] J. M. Franco: Exponentially fitted explicit Runge–Kutta–Nyström methods, *J. Comput. Appl. Math.* **167** (2004) 1–19 (see p. 40).

[78] J. M. Franco: Runge–Kutta methods adapted to the numerical integration of oscillatory problems, *Appl. Numer. Math.* **50** (2004) 427–443 (see p. 40).

[79] B. Fuchssteiner and A. S. Fokas: Symplectic structures, their Bäcklund transformations and hereditary symmetries, *Phys. D* **4** (1981) 47–66 (see p. 83).

[80] D. Furihata: Finite difference schemes for $\frac{\partial u}{\partial t} = \left(\frac{\partial}{\partial x}\right)^{\alpha} \frac{\delta G}{\delta u}$ that inherit energy conservation or dissipation property, *J. Comput. Phys.* **156** (1999) 181–205 (see pp. 7, 75, 76).

[81] D. Furihata: Discrete variational method for partial differential equation (in Japanese), PhD Thesis, The University of Tokyo, 1996 (see pp. 7, 75).

[82] D. Furihata and T. Matsuo: *Discrete Variational Derivative Method: A Structure-Preserving Numerical Method for Partial Differential Equations*. Chapman & Hall/CRC, Boca Raton, 2011 (see pp. 78, 85).

[83] D. Furihata and M. Mori: General derivation of finite difference schemes by means of a discrete variation (in Japanese), *Trans. Japan Soc. Indust. Appl. Math.* **8** (1998) 317–340 (see p. 75).

[84] L. Galgani, A. Giorgilli, A. Martinoli and S. Vanzini: On the problem of energy equipartition for large systems of the Fermi–Pasta–Ulam type: analytical and numerical estimates, *Phys. D* **59** (1992) 334–348 (see p. 25).

[85] B. García-Archilla, J. M. Sanz-Serna and R. D. Skeel: Long-time-step methods for oscillatory differential equations, *SIAM J. Sci. Comput.* **20** (1998) 930–963 (see p. 61).

[86] W. Gautschi: Numerical integration of ordinary differential equations based on trigonometric polynomials, *Numer. Math.* **3** (1961) 381–397 (see pp. 40, 61).

[87] C. Gear: Hybrid methods for initial value problems in ordinary differential equations, *SIAM J. Numer. Anal., Ser. B.* **2** (1965) 69–86 (see p. 14).

[88] O. Gonzalez: Time integration and discrete Hamiltonian systems, *J. Nonlinear Sci.* **6** (1996) 449–467 (see pp. 34, 35).

[89] W. B. Gragg and H. J. Stetter: Generalized multistep predictor-corrector methods, *J. ACM* **11** (1964) 188–209 (see p. 14).

[90] D. Greenspan: *Discrete Models*. Addison-Wesley Publishing Co., Reading, MA, 1973 (see p. 35).

[91] V. Grimm and M. Hochbruck: Error analysis of exponential integrators for oscillatory second-order differential equations, *J. Phys. A* **39** (2006) 5495 (see pp. 40, 61).

[92] E. Hairer: Symmetric projection methods for differential equations on manifolds, *BIT* **40** (2000) 726–734 (see p. 34).

[93] E. Hairer: Geometric integration of ordinary differential equations on manifolds, *BIT* **41** (2001) 996–1007 (see p. 34).

[94] E. Hairer: Energy-preserving variant of collocation methods, *J. Numer. Anal. Ind. Appl. Math.* **5** (2010) 73–84 (see pp. 36, 37, 48).

[95] E. Hairer and C. Lubich: Long-time energy conservation of numerical methods for oscillatory differential equations, *SIAM J. Numer. Anal.* **38** (2000) 414–441 (see pp. 40, 61).

[96] E. Hairer, C. Lubich and G. Wanner: Geometric numerical integration illustrated by the Störmer-Verlet method, *Acta Numerica* **12** (2003) 399–450 (see p. 3).

[97] E. Hairer, C. Lubich and G. Wanner: *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. 2nd ed. Springer-Verlag, Heidelberg, 2006 (see pp. 4, 13, 16–18, 20, 22, 23, 25, 26, 28, 33, 34, 37, 60, 61).

[98] E. Hairer, R. I. McLachlan and A. Razakarivony: Achieving Brouwer's law with implicit Runge–Kutta methods, *BIT* **48** (2008) 231–243 (see p. 55).

[99] E. Hairer, S. P. Nørsett and G. Wanner: *Solving Ordinary Differential Equations I: Nonstiff Problems*. 2nd ed. Springer-Verlag, Berlin, 1993 (see pp. 1, 4, 14, 17–19, 23).

[100] E. Hairer and G. Wanner: On the Butcher group and general multi-value methods, *Computing* **13** (1974) 1–15 (see pp. 16, 21).

[101] E. Hairer and G. Wanner: *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. 2nd ed. Springer-Verlag, Berlin, 1996 (see p. 1).

[102] E. Hairer and C. J. Zbinden: On conjugate symplecticity of B-series integrators, *IMA J. Numer. Anal.* **33** (2013) 57–79 (see p. 37).

[103] K. Heun: Neue Methoden zur approximativen Integration der Differentialgleichungen einer unabhängigen Veränderlichen, *Z. Math. Phys.* **45** (1900) 23–38 (see p. 14).

[104] M. Hochbruck and C. Lubich: A Gautschi-type method for oscillatory second-order differential equations, *Numer. Math.* **83** (1999) 403–426 (see pp. 40, 61).

[105] H. Holden, K. H. Karlsen and N. H. Risebro: Convergent difference schemes for the Hunter–Saxton equation, *Math. Comput.* **76** (2007) 699–744 (see p. 86).

[106] D. D. Holm and R. I. Ivanov: Smooth and peaked solitons of the CH equation, *J. Phys. A* **43** (2010) 434003 (see p. 85).

[107] D. D. Holm, T. Schmah and C. Stoica: *Geometric Mechanics and Symmetry: From Finite to Infinite Dimensions*. Oxford University Press, Oxford, 2009 (see p. 23).

[108] J. Hong and C. Li: Multi-symplectic Runge–Kutta methods for nonlinear Dirac equations, *J. Comput. Phys.* **211** (2006) 448–472 (see p. 82).

[109] J. Hong, X. Y. Liu and C. Li: Multi-symplectic Runge–Kutta–Nyström methods for nonlinear Schrödinger equations with variable coefficients, *J. Comput. Phys.* **226** (2007) 1968–1984 (see p. 82).

[110] W. Huang and R. D. Russell: *Adaptive Moving Mesh Methods*. Springer-Verlag, New York, 2011 (see p. 124).

[111] J. K. Hunter and R. Saxton: Dynamics of director fields, *SIAM J. Appl. Math.* **51** (1991) 1498–1521 (see pp. 84, 85).

[112] J. K. Hunter and Y. X. Zheng: On a nonlinear hyperbolic variational equation. I. Global existence of weak solutions, *Arch. Rational Mech. Anal.* **129** (1995) 305–353 (see p. 85).

[113] J. K. Hunter and Y. X. Zheng: On a nonlinear hyperbolic variational equation. II. The zero-viscosity and dispersion limits, *Arch. Rational Mech. Anal.* **129** (1995) 355–383 (see p. 85).

[114] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett and A. Zanna: Lie-group methods, *Acta Numerica* **9** (2000) 215–365 (see p. 34).

[115] Y. Ishimori: A high-order energy-conserving integration scheme for Hamiltonian systems, *Phys. Lett. A* **372** (2008) 1562–1573 (see p. 35).

[116] T. Itoh and K. Abe: Hamiltonian-conserving discrete canonical equations based on variational difference quotients, *J. Comput. Phys.* **76** (1988) 85–102 (see p. 35).

[117] T. Kawahara: Oscillatory solitary waves in dispersive media, *J. Phys. Soc. Jpn.* **33** (1972) 260–264 (see p. 83).

[118] M. Kohlmann: The curvature of semidirect product groups associated with two-component Hunter–Saxton systems, *J. Phys. A* **44** (2011) 225203 (see p. 85).

[119] R. Kozlov: Conservative discretizations of the Kepler motion, *J. Phys. A* **40** (2007) 4529 (see p. 34).

[120] H. Kuramae: An alternating discrete variational derivative method and its applications (in Japanese), Master Thesis, The University of Tokyo, 2012 (see p. 82).

[121] W. Kutta: Beitrag zur näherungsweisen Integration totaler Differentialgleichungen, *Z. Math. Phys.* **46** (1901) 435–453 (see p. 14).

[122] F. M. Lasagni: Canonical Runge–Kutta methods, *Z. Angew. Math. Phys.* **39** (1988) 952–953 (see p. 31).

[123] B. Leimkuhler and S. Reich: *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge, 2004 (see pp. 4, 13, 18, 27, 80).

[124] J. Lenells: Poisson structure of a modified Hunter–Saxton equation, *J. Phys. A* **41** (2008) 285207 (see p. 85).

[125] J. Lenells: The Hunter-Saxton equation: a geometric approach, *SIAM J. Math. Anal.* **40** (2008) 266–277 (see p. 85).

[126] J. Lenells: Periodic solitons of an equation for short capillary-gravity waves, *J. Math. Anal. Appl.* **352** (2009) 964–966 (see p. 85).

[127] J. Lenells: Spheres, Kähler geometry, and the Hunter–Saxton system, *Proc. R. Soc. A* **469** (2013) 20120726 (see p. 85).

[128] J. Lenells and M. Wunsch: The Hunter–Saxton system and the geodesics on a pseudosphere, *Comm. Partial Differential Equations* **38** (2013) 860–881 (see p. 85).

[129] Q. Li, Y. Song and Y. Wang: On multi-symplectic partitioned Runge–Kutta methods for Hamiltonian wave equations, *Appl. Math. Comput.* **177** (2006) 36–43 (see p. 82).

[130] Q. Li, Y. Sun and Y. Wang: On multisymplectic integrators based on Runge–Kutta–Nyström methods for Hamiltonian wave equations, *Appl. Math. Comput.* **182** (2006) 1056–1063 (see p. 82).

[131] T. Lyche: Chebyshevian multistep methods for ordinary differential equations, *Numer. Math.* **19** (1972) 65–75 (see p. 40).

[132] J. E. Marsden and T. S. Ratiu: *Introduction to Mechanics and Symmetry*. 2nd ed. Springer-Verlag, New York, 1999 (see pp. 23, 69).

[133] J. E. Marsden and M. West: Discrete mechanics and variational integrators, *Acta Numerica* **10** (2001) 357–514 (see p. 13).

[134] T. Matsuo: New conservative schemes with discrete variational derivatives for nonlinear wave equations, *J. Comput. Appl. Math.* **203** (2007) 32–56 (see p. 78).

[135] T. Matsuo: Dissipative/conservative Galerkin method using discrete partial derivatives for nonlinear evolution equations, *J. Comput. Appl. Math.* **218** (2008) 506–521 (see pp. 8, 78, 82, 87–90).

[136] T. Matsuo: A Hamiltonian-conserving Galerkin scheme for the Camassa–Holm equation, *J. Comput. Appl. Math.* **234** (2010) 1258–1266 (see pp. 82, 106).

[137] T. Matsuo and D. Furihata: Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations, *J. Comput. Phys.* **171** (2001) 425–447 (see pp. 35, 78).

[138] T. Matsuo and D. Furihata: A stabilization of multistep linearly implicit schemes for dissipative systems, *J. Comput. Appl. Math.* **264** (2014) 38–48 (see pp. 35, 78).

[139] T. Matsuo and H. Yamaguchi: An energy-conserving Galerkin scheme for a class of nonlinear dispersive equations, *J. Comput. Phys.* **228** (2009) 4346–4358 (see p. 82).

[140] R. I. McLachlan: On the numerical integration of ordinary differential equations by symmetric composition methods, *SIAM J. Sci. Comput.* **16** (1995) 151–168 (see p. 20).

[141] R. I. McLachlan, G. R. W. Quispel and N. Robidoux: Geometric integration using discrete gradients, *Phil. Trans. R. Soc. Lond. A* **357** (1999) 1021–1045 (see p. 35).

[142] R. H. Merson: An operational method for the study of integration processes, in *Proc. Symp. Data Processing*. 1956, 110–125 (see p. 17).

[143] Y. Minesaki and Y. Nakamura: A new discretization of the Kepler motion which conserves the Runge–Lenz vector, *Phys. Lett. A* **306** (2002) 127–133 (see p. 34).

[144] Y. Miyatake: Trigonometric methods for highly oscillatory Hamiltonian systems based on the discrete gradient method for Lagrange's equation, in preparation (see p. 8).

[145] Y. Miyatake: An energy-preserving exponentially-fitted continuous stage Runge–Kutta method for Hamiltonian systems, *BIT* **54** (2014) 777–799 (see p. 8).

[146] Y. Miyatake: A derivation of energy-preserving exponentially-fitted integrators for Poisson systems, *Comput. Phys. Commun.* **187** (2015) 156–161 (see p. 8).

[147] Y. Miyatake: Structure-Preserving Numerical Methods for Nonlinear Partial Differential Equations, Master Thesis, The University of Tokyo, 2012 (see p. 8).

[148] Y. Miyatake, D. Cohen, D. Furihata and T. Matsuo: Geometric numerical integrators for Hunter–Saxton-like equations, submitted (see p. 8).

[149] Y. Miyatake and T. Matsuo: A general framework for finding energy dissipative/conservative H^1-Galerkin schemes and their underlying H^1-weak forms for nonlinear evolution equations, to appear in BIT (see p. 8).

[150] Y. Miyatake and T. Matsuo: Conservative finite difference schemes for the Degasperis–Procesi equation, *J. Comput. Appl. Math.* **236** (2012) 3728–3740 (see p. 78).

[151] Y. Miyatake and T. Matsuo: Energy-preserving H^1-Galerkin schemes for shallow water wave equations with peakon solutions, *Phys. Lett. A* **376** (2012) 2633–2639 (see pp. 105, 107).

[152] Y. Miyatake and T. Matsuo: A note on the adaptive conservative/dissipative discretization for evolutionary partial differential equations, *J. Comput. Appl. Math.* **274** (2015) 79–87 (see pp. 8, 84, 127).

[153] Y. Miyatake, T. Matsuo and D. Furihata: Invariants-preserving integration of the modified Camassa–Holm equation, *Japan J. Indust. Appl. Math.* **28** (2011) 351–381 (see pp. 78, 85).

[154] Y. Miyatake, T. Yaguchi and T. Matsuo: Numerical integration of the Ostrovsky equation based on its geometric structures, *J. Comput. Phys.* **231** (2012) 4542–4559 (see pp. 78, 85).

[155] B. Moon and Y. Liu: Wave breaking and global existence for the generalized periodic two-component Hunter–Saxton system, *J. Differential Equations* **253** (2012) 319–355 (see p. 85).

[156] B. E. Moore: A modified equations approach for multi-symplectic integration methods, PhD Thesis, University of Surrey, 2003 (see p. 81).

[157] B. Moore and S. Reich: Backward error analysis for multi-symplectic integration methods, *Numer. Math.* **95** (2003) 625–652 (see pp. 79, 81).

[158] P. J. Olver: *Applications of Lie Groups to Differential Equations*. 2nd ed. Springer-Verlag, New York, 1993 (see pp. 69, 71).

[159] L. A. Ostrovsky: Nonlinear internal waves in the rotating ocean, *Okeanologiia* **18** (1978) 181–191 (see p. 84).

[160] K. Ozawa: A functional fitting Runge–Kutta method with variable coefficients, *Japan J. Indust. Appl. Math.* **18** (2001) 107–130 (see pp. 40, 44, 45).

[161] K. Ozawa: A functionally fitted three-stage explicit singly diagonally implicit Runge–Kutta method, *Japan J. Indust. Appl. Math.* **22** (2005) 403–427 (see pp. 40, 44).

[162] B. Paternoster: Runge–Kutta(–Nyström) methods for ODEs with periodic solutions based on trigonometric polynomials, *Appl. Numer. Math.* **28** (1998) 401–412 (see p. 40).

[163] M. V. Pavlov: The Gurevich–Zybin system, *J. Phys. A* **38** (2005) 3823–3840 (see p. 85).

[164] A. Potra and W. C. Rheinbold: On the numerical solution of Euler–Lagrange equations, *Mech. Struct. Mach.* **19** (1991) 1–18 (see p. 34).

[165] A. Potra and J. Yen: Implicit numerical integration for Euler–Lagrange equations via tangent space parametrization, *Mech. Struct. Mach.* **19** (1991) 77–98 (see p. 34).

[166] A. Preissmann: *Propagation des intumescences dans les canaux et rivières in First Congress of the French Association for Computaton*. Grenoble, 1961 (see p. 79).

[167] G. R. W. Quispel and D. I. McLaren: A new class of energy-preserving numerical integration methods, *J. Phys. A* **41** (2008) 045206 (see pp. 4, 5, 35).

[168] X. Raynaud: On a shallow water wave equation, PhD Thesis, Norwegian University of Science and Technology, 2006 (see p. 85).

[169] W. H. Reed and T. R. Hill: Triangular mesh methods for the neutron transport equation, tech. rep. Los Alamos Scientific Laboratory Report, 1973 (see p. 110).

[170] S. Reich: Multi-symplectic Runge–Kutta collocation methods for Hamiltonian wave equations, *J. Comput. Phys.* **157** (2000) 473–499 (see p. 82).

[171] C. Runge: Ueber die numerische Auflösung von Differentialgleichungen, *Math. Ann.* **46** (1895) 167–178 (see p. 14).

[172] J. M. Sanz-Serna: Methods for the numerical solution of the nonlinear Schrödinger equation, *Math. Comput.* **43** (1984) 21–27 (see p. 7).

[173] J. M. Sanz-Serna: Runge–Kutta schemes for Hamiltonian systems, *BIT* **28** (1988) 877–883 (see p. 31).

[174] T. Schäfer and C. E. Wayne: Propagation of ultra-short optical pulses in cubic nonlinear media, *Phys. D* **196** (2004) 90–105 (see p. 84).

[175] T. E. Simos: An exponentially-fitted Runge–Kutta method for the numerical integration of initial-value problems with periodic or oscillating solutions, *Comput. Phys. Comm.* **115** (1998) 1–8 (see p. 40).

[176] C. Störmer: Sur les trajectoires des corpuscules électrisés, *Arch. sci. phys. nat., Genéve* **24** (1907) 5–18, 113–158, 221–247 (see pp. 1, 31).

[177] W. Strauss and L. Vazquez: Numerical solution of a nonlinear Klein–Gordon equation, *J. Comput. Phys.* **28** (1978) 271–278 (see p. 7).

[178] Y. B. Suris: On the conservation of the symplectic structure in the numerical solution of Hamiltonian systems (in Russian), in *Numerical Solution of Ordinary Differential Equations*. Keldysh Institute of Applied Mathematics, USSR Academy of Sciences, Moscow, 1988, 148–160 (see p. 31).

[179] M. Suzuki: Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations, *Phys. Lett. A* **146** (1990) 319–323 (see p. 20).

[180] J. Swift and P. C. Hohenberg: Hydrodynamic fluctuations at the convective instability, *Phys. Rev. A* **15** (1977) 319–328 (see p. 83).

[181] K. Takeya: Conservative finite difference schemes for the Camassa–Holm equation (in Japanese), Master Thesis, Osaka University, 2007 (see pp. 78, 85).

[182] W. Tang and Y. Sun: Construction of Runge–Kutta type methods for solving ordinary differential equations, *Appl. Math. Comput.* **234** (2014) 179–191 (see p. 37).

[183] M. Van Daele and G. Vanden Berghe: Geometric numerical integration by means of exponentially-fitted methods, *Appl. Numer. Math.* **57** (2007) 415–435 (see pp. 40, 53).

[184] H. Van de Vyver: A symplectic exponentially fitted modified Runge–Kutta–Nyström method for the numerical integration of orbital problems, *New Astronomy* **10** (2005) 261–269 (see pp. 44, 46).

[185] H. Van de Vyver: A fourth-order symplectic exponentially fitted integrator, *Comput. Phys. Comm.* **174** (2006) 255–262 (see pp. 44, 46).

[186] G. Vanden Berghe, H. De Meyer, M. Van Daele and T. Van Hecke: Exponentially-fitted explicit Runge–Kutta methods, *Comput. Phys. Comm.* **123** (1999) 7–15 (see pp. 40, 44).

[187] G. Vanden Berghe, H. De Meyer, M. Van Daele and T. Van Hecke: Exponentially fitted Runge–Kutta methods, *J. Comput. Appl. Math.* **125** (2000) 107–115 (see pp. 40, 44).

[188] G. Vanden Berghe, L. G. Ixaru and H. De Meyer: Frequency determination and step-length control for exponentially-fitted Runge–Kutta methods, *J. Comput. Appl. Math.* **132** (2001) 95–105 (see p. 40).

[189] G. Vanden Berghe and M. Van Daele: Symplectic exponentially-fitted four-stage Runge–Kutta methods of the Gauss type, *Numer. Algor.* **56** (2011) 591–608 (see p. 46).

[190] G. Vanden Berghe, M. Van Daele and H. Vande Vyver: Exponential fitted Runge–Kutta methods of collocation type: fixed or variable knot points?, *J. Comput. Appl. Math.* **159** (2003) 217–239 (see pp. 40, 45).

[191] O. V. Vasilyev and S. Paolucci: A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain, *J. Comput. Phys.* **125** (1996) 498–512 (see p. 124).

[192] O. V. Vasilyev and S. Paolucci: A fast adaptive wavelet collocation algorithm for multidimensional PDEs, *J. Comput. Phys.* **138** (1997) 16–56 (see pp. 124, 126).

[193] O. V. Vasilyev, S. Paolucci and M. Sen: A multilevel wavelet collocation method for solving partial differential equations in a finite domain, *J. Comput. Phys.* **120** (1995) 33–47 (see p. 124).

[194] L. Verlet: Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard–Jones molecules, *Phys. Rev.* **159** (1967) 98–103 (see pp. 1, 31).

[195] R. de Vogelaere: Methods of integration which preserve the contact transformation property of the Hamiltonian equations, Department of Mathematics, University of Notre Dame, Report 4, 1956 (see p. 30).

[196] A. M. Wazwaz: New solitary wave solutions to the Kuramoto–Sivashinsky and the Kawahara equations, *Appl. Math. Comput.* **182** (2006) 1642–1650 (see p. 104).

[197] K. Wright: Some relationships between implicit Runge–Kutta, collocation and Lanczos τ methods, and their stability properties, *BIT* **10** (1970) 217–227 (see p. 19).

[198] H. Wu and M. Wunsch: Global existence for the generalized two-component Hunter–Saxton system, *J. Math. Fluid Mech.* **14** (2012) 455–469 (see p. 85).

[199] X. Wu, B. Wang and W. Shi: Efficient energy-preserving integrators for oscillatory Hamiltonian systems, *J. Comput. Phys.* **235** (2013) 587–605 (see p. 40).

[200] M. Wunsch: On the Hunter–Saxton system, *Discrete Contin. Dyn. Syst. Ser. B* **12** (2009) 647–656 (see p. 85).

[201] M. Wunsch: The generalized Hunter–Saxton system, *SIAM J. Math. Anal.* **42** (2010) 1286–1304 (see p. 85).

[202] Y. Xia, Y. Xu and C.-W. Shu: Local discontinuous Galerkin methods for the Cahn–Hilliard type equations, *J. Comput. Phys.* **227** (2007) 472–491 (see pp. 110, 121).

[203] Y. Xing, C. S. Chou and C.-W. Shu: Energy conserving local discontinuous Galerkin methods for wave propagation problems, *Inverse Problems and Imaging* **7** (2013) 967–986 (see p. 110).

[204] Y. Xu and Shu: Local discontinuous Galerkin methods for nonlinear Schrödinger equations, *J. Comput. Phys.* **205** (2005) 72–97 (see p. 110).

[205] Y. Xu and C.-W. Shu: A local discontinuous Galerkin method for the Camassa–Holm equation, *SIAM J. Numer. Anal.* **46** (2008) 1998–2021 (see p. 110).

[206] Y. Xu and C.-W. Shu: Local discontinuous Galerkin method for the Hunter–Saxton equation and its zero-viscosity and zero-dispersion limits, *SIAM J. Sci. Comput.* **31** (2009) 1249–1268 (see p. 86).

[207] Y. Xu and C.-W. Shu: Dissipative numerical methods for the Hunter–Saxton equation, *J. Comput. Math.* **28** (2010) 606–620 (see p. 86).

[208] T. Yaguchi: A Lagrangian approach to deriving energy-preserving numerical schemes for the Euler–Lagrange partial differential equations, *Math. Model. Numer. Anal.* **47** (2013) 1493–1513 (see p. 38).

[209] T. Yaguchi, T. Matsuo and M. Sugihara: An extension of the discrete variational method to nonuniform grids, *J. Comput. Phys.* **229** (2010) 4382–4423 (see p. 78).

[210] T. Yaguchi, T. Matsuo and M. Sugihara: Conservative numerical schemes for the Ostrovsky equation, *J. Comput. Appl. Math.* **234** (2010) 1036–1048 (see pp. 78, 85).

[211] T. Yaguchi, T. Matsuo and M. Sugihara: The discrete variational derivative method based on discrete differential forms, *J. Comput. Phys.* **231** (2012) 3963–3986 (see p. 78).

[212] J. Yan and C.-W. Shu: A local discontinuous Galerkin method for KdV type equations, *SIAM J. Numer. Anal.* **40** (2002) 769–791 (see pp. 110, 121).

[213] K. Yee: Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, *IEEE Trans. Antennas Propag.* **14** (1966) 302–307 (see p. 7).

[214] N. Yi, Y. Huang and H. Liu: A direct discontinuous Galerkin method for the generalized Korteweg–de Vries equation: energy conservation and boundary effect, *J. Comput. Phys.* **242** (2013) 351–366 (see pp. 110, 121).

[215] H. Yoshida: Construction of higher order symplectic integrators, *Phys. Lett. A* **150** (1990) 262–268 (see p. 20).

[216] P. F. Zhao and M. Z. Qin: Multisymplectic geometry and multisymplectic Preissmann scheme for the KdV equation, *J. Phys. A* **33** (2000) 3613–3626 (see p. 73).

[217] G. Zhong and J. E. Marsden: Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators, *Phys. Lett. A* **133** (1988) 134–139 (see p. 4).

[218] H. Zhu, L. Tang, S. Song, Y. Tang and D. Wang: Symplectic wavelet collocation method for Hamiltonian wave equations, *J. Comput. Phys.* **229** (2010) 2550–2572 (see p. 126).

[219] M. G. Zielonka, M. Ortiz and J. E. Marsden: Variational r-adaption in elastodynamics, *Internat. J. Numer. Methods Engrg.* **74** (2008) 1162–1197 (see p. 84).