Doctoral theses at NTNU, 2020:166

Sølve Eidnes

Invariant-preserving integrators for differential equations

Doctoral thesis

Sølve Eidnes

Doctoral theses at NTNU, 2020:166

NTNU
Norwegian University of Science and Technology
Thesis for the Degree of
Philosophiae Doctor
Faculty of Information Technology and Electrical
Engineering
Department of Mathematical Sciences

ISBN 978-82-326-4680-7 (printed ver.)
ISBN 978-82-326-4681-4 (electronic ver.)
ISSN 1503-8181

NTNU
Norwegian University of
Science and Technology

NTNU

NTNU
Norwegian University of
Science and Technology

Sølve Eidnes

Invariant-preserving integrators for differential equations

Thesis for the Degree of Philosophiae Doctor

Trondheim, June 2020

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

ISBN 978-82-326-4680-7 (printed ver.)
ISBN 978-82-326-4681-4 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2020:166

Printed by NTNU Grafisk senter

Preface

This thesis is submitted in partial fulfilment of the requirements for the degree of philosophiae doctor (PhD) at the Norwegian University of Science and Technology (NTNU). Financed by the RCN research project "Structure preserving integrators, discrete integrable systems and algebraic combinatorics", the work has been carried out at the Department of Mathematics, NTNU. The EU Horizon 2020 project "Challenges in preservation of structure" has provided additional funding for travels, including a research stay at the University of Cambridge in the autumn semester of 2017.

I have thoroughly enjoyed my four years as a PhD student. I am very grateful to my supervisor Brynjulf Owren for giving me this opportunity, and for his invaluable guidance and support along the way. Many thanks also go to my co-supervisor Elena Celledoni for her enthusiastic and insightful supervision.

I have been fortunate to be involved in many interesting and rewarding projects, which has resulted in a variety of papers. I thank my co-authors Markus Eslitzbichler, Lu Li, Torbjørn Ringholm, Shun Sato and Alexander Schmeding for all their hard work. I am especially grateful to Torbjørn and Lu, whom I have collaborated most closely with. Working with them has without doubt enhanced both the quality of my work and my enjoyment of doing it.

As much as I have liked doing research, my favourite part of the work day has usually been the lunch breaks. I thank my good friends and colleagues at the Department of Mathematics for that.

Lastly I thank my parents, Rannveig and Bjørn Inge, and my siblings, Knut and Åse, for their encouragement and support.

Sølve Eidnes
Trondheim, April 29, 2020

Contents

Introduction

Numerical integrators are methods for solving differential equations, which usually cannot be solved exactly. Such methods are used daily by researchers in a broad range of sciences and by engineers in the industry. The development and study of new and improved numerical integrators is thus one of the most important fields in applied mathematics, reflected in its long history and the vast research being done in the field today.

The performance of a numerical integrator is typically measured by its ability to approximate the exact solution as accurately as possible. This ability has historically been judged by calculating the numerical error at a given time and weighing that against the computational cost. This is a *quantitative* measurement of the error. However, as the capabilities of computers to perform demanding operations improve, so follows an increased demand for numerical methods that perform well over very long time intervals. This can in part explain the growing interest in *geometric integrators* over the last few decades. Such integrators are developed based on their *qualitative* behaviour, i.e. whether structures of the system it models is also present in the numerical solution. This evolution marks a shift from most of the emphasis being on general-purpose algorithms to an increased interest in special-purpose methods, developed for classes of differential equations with common characteristics.

Over the last three decades, methods have been constructed that preserve structures such as symplecticity [67], symmetry [52], phase-space volume [62], invariants and dissipativity [51], and Lie group structures [35]. The focus on geometric integrators has also revealed the structure-preserving properties of many classical methods. For instance, the good performance of the leapfrog scheme can in part be explained by it being a symplectic integrator [24]. Among the extensive literature covering numerical geometric integration, we recommend [3, 31, 44, 67].

In Figure 1, we plot two different numerical solutions of a double well system with a conserved Hamiltonian: one obtained by the explicit midpoint method and one obtained by an invariant-preserving method. Both these methods give second order approximations of the exact solution. After a very long integration time with a fairly large step size we cannot assume that either of the methods yield accurate approximations of the exact solution, but a method preserving the Hamiltonian exactly is guaranteed to stay on the closed path of the exact solution. The explicit midpoint method is much faster than the implicit invariant-preserving method, but the plots in Figure 1 indicate at least three obvious advantages of using a geometric integrator. For one, while the solution obtained by the explicit midpoint midpoint veers increasingly far from the exact solution, there is a bound to how inaccurate the solution of the invariant-preserving scheme can be. Related to this, the non-preserving method will eventually become unstable, as we see indications of in the plot. Lastly, there may very well be situations where staying on the correct solution path is important in itself, in which case an invariant-preserving scheme is clearly advantageous.

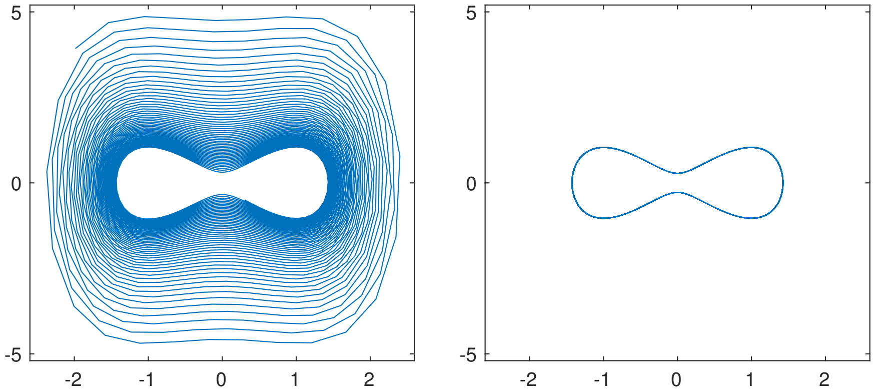


Figure 1: A double well system solved by the explicit midpoint method (left) and an invariant-preserving discrete gradient method (right). In both cases, the step size $h = 0.1$ and the end time $T = 250$.

This thesis is a collection of nine papers, of which eight are written in collaboration with others. The title of the thesis refers both specifically to the subject of the majority of the papers, as well as in a broader sense to the focus of the whole thesis, depending on how one defines *invariant*. The term invariant is often used interchangably with first integrals, which we will define later. Then an invariant is meant to be a function that remains unchanged over time along the solution of a differential equation. Numerical methods preserving such first integrals are called energy-preserving methods. Among these methods are the discrete gradient methods, which features in the first six papers of this thesis. When used to solve systems with first integrals, these methods preserve the first integral to machine precision. However, in the fourth paper, we do not study the energy-preserving property of discrete gradient methods, but rather employ them to gradient flow problems. Then the methods are preserving in the sense that the numerical solution preserves the dissipative behaviour of the continuous system it models.

Kahan's method is considered in papers 6 and 7. In contrast to the discrete gradient methods, this does not preserve a first integral to machine precision. We still use the term energy-preserving about this method, since it keeps the error of the first integral within a certain bound by preserving a modified first integral exactly. The two last papers are on the surface only loosely related to the rest of the thesis; they are within the field of shape analysis, and not concerning the preservation of first integrals. There we study invariant preservation in a different setting: we present methods for matching curves in various manifolds, where this matching is invariant under changing parametrization. The parametrized curves are mapped to tangent vector fields where geodesics are computed. The solution of differential equation comes into play in this process when the curves are mapped back to the manifold.

In the remainder of this introduction we will present some of the concepts most frequently used in the papers to follow, leaving most of the details for later. We also discuss our motivation for undertaking the different tasks, and aim to put the research in a larger context. Lastly we give a brief summary of each of the papers.

Preservation of first integrals

Consider the ordinary differential equation (ODE) system

$$\dot{x} = f(x), \quad x(t_0) = x_0, \quad x \in \mathbb{R}^d, f : \mathbb{R}^d \to \mathbb{R}^d. \tag{1}$$

Such a system may have one or more *first integrals*: functions $H : \mathbb{R}^d \to \mathbb{R}$ such that $H(x(t)) = H(x(t_0))$ for any $t > t_0$. First integral is one of many names used for such a function; others include *invariant*, *conserved quantity*, *constant of motion*, or *energy*, even when it is not the energy of the system in the physical sense. In this thesis, we mainly call it energy or first integral, and we call methods preserving it energy-preserving or integral-preserving methods. If

$$f(x)^T \nabla H(x) = 0, \tag{2}$$

then (1) preserves the energy H, since

$$\frac{\mathrm{d}}{\mathrm{d}t} H(x) = \nabla H(x)^T \dot{x} = 0. \tag{3}$$

If we can write

$$f(x) = S(x) \nabla H(x), \tag{4}$$

where $S(x) : \mathbb{R}^{d \times d} \to \mathbb{R}^d$ is a skew-symmetric matrix, then (1) preserves H: this follows from the skew-symmetry of $S(x)$, which yields

$$\nabla H(x)^T f(x) = \nabla H(x)^T S(x) \nabla H(x) = 0. \tag{5}$$

The converse is also true, on $\{x \in \mathbb{R}^d : \nabla H(x) \neq 0\}$; in [51], McLachlan et al. show that if H is a first integral of (1), then there exists a skew-symmetric matrix $S(x)$, bounded near every non-degenerate critical point of H, such that $f = S(x)\nabla H(x)$. They show this by providing an explicit expression for one such $S(x)$: the so-called default formula

$$S(x) = \frac{f(x)\nabla H(x)^T - \nabla H(x) f(x)^T}{\nabla H(x)^T \nabla H(x)}. \tag{6}$$

Many ODEs are well-known to have first integrals, and are often formulated directly on the so-called *skew-gradient form*

$$\dot{x} = S(x)\nabla H(x). \tag{7}$$

This includes the large class of canonical Hamiltonian ODEs, in which case S is a constant matrix of a certain form, and non-canonical Hamiltonian systems, in which case S can depend on the solution [31,44].

As mentioned earlier, many methods have preservation properties even if they were not constructed specifically for that cause. This is the case for Runge–Kutta methods and first integrals: all Runge–Kutta methods preserve linear first integrals, and a large class preserves quadratic invariants [20,31]. However, for $n \geq 3$, no Runge–Kutta method, explicit or implicit, can preserve all polynomial invariants of degree n [31]. This has motivated the study of methods specifically designed to preserve any first integral. Among these are projection methods: we define then the submanifold of $\mathbb{R}^d$ where H is preserved, $\mathcal{M} = \{x : H(x) = H(x_0)\}$, and project our solution onto this manifold after solving it with an arbitrary one-step method in $\mathbb{R}^d$ [32, section VII.2]. Another class of energy-preserving methods was first formulated by Gonzalez in [28], and has since been the subject of many papers, including the majority of those that constitute this thesis: *discrete gradient methods*.

Discrete gradient methods

The idea behind discrete gradient methods is to find consistent discrete approximations $S(x, y, h)$ and $\overline{\nabla} H(x, y)$ to $S(x)$ and $\nabla H(x)$ in the system (7), defined such that the numerical scheme

$$\frac{x^{n+1} - x^n}{h} = \overline{S}(x^n, x^{n+1}, h)\overline{\nabla} H(x^n, x^{n+1}) \tag{8}$$

inherits the energy preservation property of the continuous system it models. The key to this is requiring that the function $\overline{\nabla} H$ satisfies

$$\overline{\nabla} H(x, y)^T (y - x) = H(y) - H(x), \tag{9}$$

a discrete analogue to (3). Together with the consistency criterion $\overline{\nabla} H(x,x) = \nabla H(x)$, this defines the *discrete gradient* (9) [51]. Requiring that $\overline{S}(x,y,h)$ is skew-symmetric and that $\overline{S}(x,x,0) = S(x)$, we have then the discrete gradient method (8). Because

$$H(x^{n+1}) - H(x^n) = h\overline{\nabla} H(x^n, x^{n+1})^T \overline{S}(x^n, x^{n+1}, h)\overline{\nabla} H(x^n, x^{n+1}) = 0,$$

this method ensures $H(x^n) = H(x(t_0))$ for any $n > 0$.

The property (9) for defining the discrete gradient is first formulated in [28], which is therefore generally regarded as the introduction of the discrete gradient method. Gonzalez called the function *discrete derivative*, and proved the corresponding schemes' ability to preserve Hamiltonians. Quispel and Turner generalized in [64] this to all systems with first integrals, i.e. systems on the skew-gradient form (7), and labeled it the discrete gradient method. It is also worth noting that schemes that rely on some discrete analogue of the property (5), although not formulated as discrete gradient methods, had appeared before the above mentioned references; see e.g. [18,36,40,45].

The choice of the discrete gradient in (8) is not necessarily unique. In fact, it is unique only if $d = 1$, and if $d > 1$ there are infinitely many functions $\overline{\nabla} H$ satisfying (9). Four different explicitly defined discrete gradients are considered in this thesis. The first, called *Gonzalez' midpoint* discrete gradient, was introduced together with the discrete gradient method in [28], and is defined by

$$\overline{\nabla}_{\mathrm{M}} H(x,y) := \nabla H\left(\frac{x+y}{2}\right) + \frac{H(y) - H(x) - \nabla H\left(\frac{x+y}{2}\right)^T (y-x)}{(y-x)^T(y-x)} \left(y - x\right).$$

In their 1988 paper [36], Itoh and Abe introduced what is later recognized as a discrete gradient, defined by

$$\overline{\nabla}_{\mathrm{IA}} H(x,y) := \sum_{j=1}^{d} c_j e_j, \tag{10}$$

where e_j is the j-th canonical unit vector and

$$c_j = \begin{cases} \dfrac{H(w_j) - H(w_{j-1})}{y_j - x_j} & \text{if } y_j \neq x_j, \\ \frac{\partial H}{\partial x_j}(w_{j-1}) & \text{if } y_j = x_j, \end{cases}$$
$$w_j = \textstyle\sum_{i=1}^{j} y_i e_i + \sum_{i=j+1}^{n} x_i e_i.$$

The advantage of this discrete gradient is two-fold: it can be computed without knowledge of the gradient, and in some cases it provides for a less expensive scheme to compute than other discrete gradients. It is however only a first order approximation of $\nabla H(x)$. We do sometimes consider a symmetrized Itoh–Abe discrete gradient $\overline{\nabla} H_{\mathrm{SIA}}(x,y) := \frac{1}{2}\left(\overline{\nabla}_{\mathrm{IA}} H(x,y) + \overline{\nabla}_{\mathrm{IA}} H(y,x)\right)$, which is of second order.

The *average vector field* (AVF) discrete gradient, sometimes called the mean-value discrete gradient, has a history dating back longer than discrete gradient methods and any other known discrete gradient [33]. It is given by the average of ∇H on the segment $[x,y]$:

$$\overline{\nabla}_{\mathrm{AVF}} H(x,y) := \int_0^1 \nabla H((1-\xi)x + \xi y)\,\mathrm{d}\xi.$$

When S in (7) is constant, the discrete gradient method with $\overline{S}(x,y,h) = S$ and $\overline{\nabla} H = \overline{\nabla}_{\mathrm{AVF}} H$ coincides with the scheme

$$\frac{x^{n+1} - x^n}{h} = \int_0^1 f((1-\xi)x^n + \xi x^{n+1})\,\mathrm{d}\xi.$$

This is sometimes viewed as a method by itself, applicable to any system (1), in which case it is called the average vector field method [63], or the AVF method. This method, which is a B-series method [14], has been generalized to a collocation-type method in [30], and its application to the time-integration of Hamiltonian partial differential equations (PDEs) has been studied extensively [11,27].

The fourth discrete gradient we consider has perhaps not been formally defined before Paper 5 of this thesis. It is the discrete gradient that, when used for the time-integration of PDEs, may yield the *discrete variational derivative method* introduced by Furihata in [25], see also [26,49,72]. In the first paper of this thesis, we show the connection between the method of Furihata and discrete gradient methods.

Any discrete gradient $\overline{\nabla} H(x,y)$ is restricted by the definition (9) to be at best a second order approximation of $\nabla H(x)$ [51]. The discrete skew-symmetric matrix $S(x,y,h)$ is often defined independent of the step-size h, including in most of the papers presented here. In that case the scheme (8) cannot quarantee higher than second order convergence to the exact solution. Approximations of $S(x)$ that render possible higher than second order convergence have been studied by McLaren and Quispel [53, 54] and by Norton et al. [59, 60]. In addition, Quispel and McLaren suggested in [63] a fourth order generalization of the AVF method which can be viewed as a discrete gradient method for (7) with constant S. To our knowledge, no theory has previously been developed for arbitrary order discrete gradient methods for all systems of the form (7). This is what we introduce in Paper 5.

Dissipative systems

If we replace the skew-symmetrix matrix $S(x)$ in (7) with a negative definite matrix $N(x)$, we have a dissipative system instead of a conservative system. That is, if

$$\dot{x} = N(x)\nabla H(x), \quad x(t_0) = x_0, \tag{11}$$

where $v^T N(x) v < 0$ for any non-zero $v \in \mathbb{R}^d$, then H is continuously decreasing towards a minimum, since $H(x(t)) \leq H(x_0)$ for any $t > 0$, with equality only if $\nabla H(x_0) = 0$. The dissipativity of this system is a structure that one may wish to preserve in the numerical solution. As first noted by McLachlan et al. in [51], the energy preservation property of discrete gradient methods is easily extended to dissipation in the numerical solution of a system (11): if we approximate $N(x)$ by a negative definite matrix $\overline{N}(x, y, h)$ such that $\overline{N}(x, x, 0) = N(x)$, then the scheme

$$\frac{x^{n+1} - x^n}{h} = \overline{N}(x^n, x^{n+1}, h)\overline{\nabla} H(x^n, x^{n+1}) \tag{12}$$

guarantees that $H(x^n) \leq H(x_0)$, with equality only if $\nabla H(x_0) = 0$.

One system that can be written on the form (11) is the gradient flow of $H(x)$, in which case $N(x) = -I$, with I being the identity matrix. Optimization problems can be formulated as gradient flow problems, with $H(x)$ or $-H(x)$ being the function one tries to find the minimum of maximum value of, respectively. When solving an optimization problem using a discrete gradient method, the Itoh–Abe discrete gradient (10) comes with an advantage: because of the iterative nature of its definition, the system (12) of d equations may be solved one scalar equation at a time. This can significantly reduce the computational cost if the alternative is to solve a coupled system of d equations, which will generally be the case if a different discrete gradient is used. Moreover, the fact that the Itoh–Abe discrete gradient is only a first order approximation of the gradient is not usually much of a disadvantage in an optimization setting, when modeling the continuous system accurately is less important than finding the optimum quickly. The use of discrete gradient methods in optimization has been presented and analysed in the papers [23,29,65,66]. With these studies being devoted to optimization in Euclidean space only, we consider an extension to Riemannian manifolds in Paper 4 of this thesis.

Solving energy-preserving PDEs

PDEs, like ODEs, may possess invariants. We consider in this thesis PDEs that can be written on the form

$$u_t = S(x, u^J)\frac{\delta \mathcal{H}}{\delta u}[u] \tag{13}$$

where the operator S is skew-symmetric with respect to the L^2 inner product, u^J denotes u and all partial derivatives with respect to spatial dimensions, and $\frac{\delta \mathcal{H}}{\delta u}[u]$ denotes the variational derivative of the functional $\mathcal{H}[u]$. The above PDE preserves the functional $\mathcal{H}$ in the sense that $\mathcal{H}[u(t)] = \mathcal{H}[u(t_0)]$ for any $t > t_0$:

$$\frac{\mathrm{d}\mathcal{H}}{\mathrm{d}t} = \left\langle \frac{\mathcal{H}}{\delta u}[u], \frac{\partial u}{\partial t} \right\rangle_{L^2} = \left\langle \frac{\mathcal{H}}{\delta u}[u], S(x, u^J) \frac{\delta \mathcal{H}}{\delta u}[u] \right\rangle_{L^2} = 0.$$

We may call the invariant $\mathcal{H}$ by different names such as first integral or energy. The connection between the above and energy-preserving ODEs is immediate: by discretizing the system (13) in the appropriate manner, one obtains a system of ODEs on the form (7).

Among the PDEs that have invariants is the important class of Hamiltonian PDEs, which can be written as (13) with $\mathcal{H}$ being a Hamiltonian and with an operator S that defines a Poisson bracket, i.e. it satisfies the Jacobi identity [61]. A well-known example that we will return to in several of the papers is the Korteweg–de Vries (KdV) equation:

$$u_t + \eta u u_x + \gamma^2 u_{xxx} = 0, \tag{14}$$

where $\eta, \gamma \in \mathbb{R}$ are constants. This equation has the two distinct Hamiltonians

$$\mathcal{H}_1[u] = \int_{\mathbb{R}} \frac{1}{2}\gamma^2 u_x^2 - \frac{1}{6}\eta u^3 \mathrm{d}x, \qquad \mathcal{H}_2[u] = \frac{1}{2}\int_{\mathbb{R}} u^2 \mathrm{d}x,$$

where $\mathcal{H}_1[u]$ is the energy of the system and $\mathcal{H}_2[u]$ is the momentum. The skew-symmetric operators associated to these integrals are

$$S_1 = \frac{\partial}{\partial x} \quad \text{and} \quad S_2 = -\frac{1}{3}\eta(\partial u + u\partial) - \gamma^2 \frac{\partial^3}{\partial x^3},$$

respectively. We note however that the KdV equation forms a completely integrable system with an infinite number of preserved integrals, for smooth solutions. Thus there are infinite different ways it can be written on the form (13).

After the increased interest and development in geometric integrators for finite-dimensional systems in the 1990s, one natural follow-up was how to generalize these methods to infinite-dimensional systems, with extra emphasis on Hamiltonian PDEs. While the phase space of a PDE is of infinite dimension, a numerical solution will be of finite dimension. Thus a numerical integrator for (13) rely on a finite dimensional approximation of the integral $\mathcal{H}$. For that reason it is necessary to clarify what is meant by an energy-preserving integrator for a PDE. In the papers to follow, we say that a numerical scheme preserves the invariant exactly if it preserves some discrete approximation exactly at every time step.

Two different but related classes of methods that have been widely studied are both considered in this thesis. The first consist of a straightforward spatial discretization of the the continuous system to obtain a system of Hamiltonian ODEs, for which a geometric integrator may be applied to get a fully discrete system. We mainly consider schemes where that integrator is a discrete gradient method. The first to study such schemes were Furihata, Matsuo and collaborators in a number of papers [47, 48, 50] and the monograph [26], building on the so-called discrete variational derivative methods first introduced in [25]. As we show in the first paper of this thesis, that method is equivalent to spatially discretizing the PDE and apply a certain discrete gradient method on the resulting system of ODEs. This alternative approach gained popularity following a paper by Celledoni et al. [11], which considers the AVF method applied to systems with a constant skew-symmetric or negative definite operator. We also note that Dahlby and Owren in [22] consider a third approach from which one can obtain the same schemes as with the other two approaches: they apply a discrete gradient method directly on the continuous system to obtain a spatially continuous method that preserves the exact Hamiltonian, deferring the spatial discretization.

The other class of methods for Hamiltonian PDEs we consider here are the multi-symplectic integrators introduced by Bridges [4, 5], and developed further by Bridges, Reich and Marsden et al. [6, 44, 46]. The idea behind these methods is to decompose the symplectic structure of Hamiltonian PDEs into independent components representing time and space. By reformulating the PDE into a multi-symplectic form, one may consider three local conservation laws: the multi-symplectic conservation law, the energy conservation and the momentum conservation law. The local conservation laws are, in contrast to global conservation laws, not dependent on the choice of boundary conditions. Thus methods preserving a discrete approximation of one or more local conservation laws have a wider area of application. When periodic boundary conditions are present, local conservation will in any case lead to global conservation. We develop a multi-symplectic integrator with the time-stepping being performed by Kahan's method in Paper 7 of this thesis.

Adaptive methods

When using PDEs to model physical problems from science or engineering, we are often required to consider a large domain even if most of the action is occurring on a small area within a given time frame. This has motivated the development of schemes with a spatial discretization that changes with time, depending on the solution parameters. We call such methods adaptive methods, or moving mesh methods. Considerable research has been undertaken to develop and study such methods in a variety of settings; it is an expansive field from which we refer to the papers [1,7,8,42,75] and the book by Huang and Russel [34], which provides a thorough treatment of the subject.

The focus of the first two papers of this thesis is a merger of adaptive methods and energy-preserving methods for PDEs. Specifically, we consider r-adaptivity, where the number of degrees of freedom are kept fixed, and combine it with the discrete gradient method for the time-stepping. Although our focus is on r-adaptivity, the approach we present is easily extended to other modes of mesh adaptivity. Our motivation for these papers sprung from the observation that many of the situations where adaptive methods seem especially useful are modeling of physical phenomena with conservation laws. Furthermore, the increased stability one hopes to gain from using an energy-preserving method may be extra important when more complexity is added to the system through adaptivity.

Prior to our contribution, energy-preserving methods for PDEs presented in the literature were almost exclusively on fixed and uniform spatial grids. Exceptions to this are two different discrete variational derivative methods on fixed, non-uniform grids, specifically defined for certain classes of PDEs [71,73], as well as an adaptive energy-preserving schemes for the KdV and Cahn–Hilliard equations developed by Miyatake and Matsuo [56].

Numerical integration on Riemannian manifolds

Papers 3 and 4 are devoted to the generalization of discrete gradient methods to Riemannian manifolds. In Paper 3 we consider the conservative case (7), while in Paper 4 the subject is dissipative systems (11). In both cases, we reformulate the ODE so that the solution evolves on the manifold M:

$$\dot{u} = S(u)\,\mathrm{grad} H(u), \quad u(0) = u^0, \quad u \in M,$$

where grad denotes the gradient defined by the Riemannian metric, and $S(u): TM \to TM$ is a tensor field having either a skew-symmetric or a dissipative structure.

The development and study of numerical methods for solving differential equations on manifolds had, like the field of geometric numerical integration as a whole, an upswing in popularity in the 1990s. By using integrators that operate directy on the manifold one avoids to either rely on local coordinates or an embedding of the manifold in a larger Euclidean space. The former approach requires a mapping between coordinate charts which will introduce extra inaccuracy to the calculations, while the latter will typically lead to numerical solutions deviating from the manifold. So far, most attention has been directed towards Lie group integrators, with the methods of Grouch and Grossmann [21] and Munthe-Kaas [57,58] being prime examples of this; see also [12,35]. Integrators on Riemannian manifolds are less common, with an exception being a generalization of the leap-frog method by Leimkuhler and Patrick [43].

The papers we present here, introducing the *discrete Riemannian gradient methods*, are largely extensions of the earlier paper [17] by Celledoni and Owren, where they generalized the discrete gradient methods to a broad class of manifolds, with a particular focus on Lie groups. We develop this further to Riemannian manifolds, whose structure provides for an intrinsic definition of the gradient and a means to measure the error of a numerical method, as well as a canonical choice of mapping between the manifold and the tangent space through the exponential and logarithmic maps.

Also seeking to develop higher order energy-preserving methods on general Riemannian manifolds, which to our knowledge had not been done before, we generalized the collocation-like method of Hairer and Cohen [19,30] in Paper 3. Their method builds on the AVF discrete gradient method, but it is in itself not a discrete gradient method. We also discuss achieving higher order methods by a composition strategy. This work was undertaken before the higher order discrete gradient methods of Paper 5 were developed, and thus a generalization of those to general manifolds is not discussed.

Since Riemannian manifolds are equipped with an Riemannian metric and through that a definition of the gradient, we have gradient flow problems occurring naturally. This gave us a motivation for applying the discrete Riemannian gradient method to solve optimization problems, which resulted in Paper 4. Further motivated by recent studies on the usage of the Itoh–Abe discrete gradient method for variational image regulation models [29,66], we developed the Itoh–Abe discrete Riemannian gradient and applied this to problems of manifold valued image denoising.

Kahan's method and linearly implicit schemes

The method of Kahan is used to solve quadratic ODEs

$$\dot{x} = Q(x) + Bx + c, \quad x \in \mathbb{R}^d,$$

where $Q(y)$ is an $\mathbb{R}^d$ valued quadratic form, $B \in \mathbb{R}^{d\times d}$ is a constant matrix, and $c \in \mathbb{R}^d$ is a constant vector. It was introduced by Kahan in the 1990s [38], and is given by

$$\frac{x^{n+1} - x^n}{h} = \overline{Q}(x^n, x^{n+1}) + B\frac{x^n + x^{n+1}}{2} + c, \tag{15}$$

where

$$\overline{Q}(x,y) := \frac{1}{2}\big(Q(x+y) - Q(x) - Q(y)\big).$$

A crucial feature of Kahan's method is that is is *linearly implicit*. That is, each term of (15) is linear in its implicitly given variable x^{n+1}. This can yield significantly shorter computational time than a fully implicit method with nonlinear expressions of x^{n+1}, e.g. if the linear system can be solved by a direct method. A system of non-linear equations is typically solved by an iterative solver where a linear system is solved at each iteration. A linearly implicit method only requires one such iteration at each time step. The development and analysis of linearly implicit schemes with preservation properties has attracted increased interest over the last few years, see e.g. [9, 37, 70, 74].

Celledoni et al. have written a series of papers investigating the geometric properties of Kahan's method [13, 15, 16]. One of these properties is its ability to preserve a modification of the invariant H if applied to a system (7) with S constant and H cubic. This modified energy is given by

$$\tilde{H}(x) = H(x) + \frac{1}{3}h\nabla H(x)^T(I - \frac{1}{2}hS\nabla^2 H(x))^{-1}S\nabla H(x).$$

So far, studies on Kahan's method has mostly been on its application to ODEs, with a notable exception being an early paper by Kahan himself together with Li [39]. In the sixth and seventh papers of this thesis, we apply Kahan's method for the time-integration of PDEs with a cubic Hamiltonian. Part of our motivation for this work was grounded in the fact that numerical schemes for PDEs of the form (13) can only hope to preserve an approximation to the integral $\mathcal{H}$. Say we discretize the PDE in space and obtain an ODE system of the form (7), with H being a discrete approximation to $\mathcal{H}$. Applying Kahan's method to this system of ODEs, it will preserve the modification $\tilde{H}$ of H, which again ensures that the error in H is bounded. While this property is obviously inferior to preserving H exactly for ODE systems, this is not so apparent for the solution of PDEs, where there in any case will be some error coming from the spatial approximation of $\mathcal{H}$.

In Paper 6, we compare Kahan's method to a different linearly implicit method with the ability to preserve a modification of a cubic H in (7). This method is a multi-step variant of the discrete gradient method, introduced by Matsuo and Furihata [48] and formalised by Dahlby and Owren in [22], in a PDE setting. We formulate it as a method for ODEs, and call it the *polarized discrete gradient method*. It is designed to give linearly implicit $(p-1)$-step schemes for systems (7) where H is a polynomial of degree p.

Shape analysis

The last two papers of this thesis are within the field of *shape analysis*. This field concerns the recognition and matching of geometric shapes, and was originally developed for planar curves. It is an area of research that has grown significantly in popularity since the turn of the century, motivated in large part by the increasing ability of computers to perform computationally demanding tasks. This has prompted the expansion of shape analysis to a variety of new areas, from higher dimensional curves to surfaces, character motions and various digital objects [2,41,55,68].

We consider in our papers shapes which are unparametrized curves evolving on a vector space, a Lie group, or a manifold. We employ the so-called *square root velocity transform* [69] to map the curves to tangent vector fields along them. Then we compare these transformed curves by computing geodesics in the L^2 metric.

The two papers included here form a continuation of the paper [10] by Celledoni et al. The connection of shape analysis to the rest of the thesis lies partly in the way our method is based on a curve transformation which is invariant under changing parametrization. Furthermore, the shape spaces we consider are Riemannian manifolds, with the intrinsic metric providing the necessary tools to compare and analyse the shapes, which establishes a connection between papers 8 and 9 and papers 3 and 4 of the thesis.

Summary of papers

PAPER 1: Adaptive energy preserving methods for partial differential equations

by *Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm*
Published in *Advances in Computational Mathematics 44 (3), pp. 815–839 (2018)*

In this paper we develop a framework for constructing adaptive methods for PDEs on the form (13) that can preserve an approximation to the invariant. The preservation is based on using a discrete gradient method for the time-stepping. We consider spatial discretization both by a finite difference approach and by partition of unity methods, which includes finite element methods. We also devote some space to explaining how discrete gradient methods applied to PDEs relate to the discrete variational derivative method of Furihata, as well as to linear projection methods. Schemes and numerical results are presented for the sine-Gordon and KdV equations.

PAPER 2: Energy preserving moving mesh methods applied to the BBM equation

by *Sølve Eidnes and Torbjørn Ringholm*
Published in *Proceedings of MekIT '17, pp. 121–136 (2017)*

This conference proceeding builds on Paper 1. We apply the adaptive and energy-preserving method developed there to the Benjamin–Bona–Mahoney equation. This PDE has exactly three conservation laws; we develop two schemes preserving different energies, and compare the numerical results. A challenge addressed here and not in the previous paper is how to treat third derivatives in the skew-symmetric operator $S(x, u^J)$ of (13) when using the finite element method.

PAPER 3: Energy preserving methods on Riemannian manifolds

by *Elena Celledoni, Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm*
Published in *Mathematics of Computation 89 (322), pp. 699–716 (2020)*

This paper, together with Paper 4, introduces the discrete Riemannian gradient methods, an extension of the discrete gradient methods (8) to finite-dimensional Riemannian manifolds. We also present accompanying generalizations of the AVF, Gonzalez' midpoint and Itoh–Abe discrete gradients, as well as higher order energy-preserving methods on Riemannian manifolds, based on composition and collocation strategies. Local and gloval error bounds for the methods are derived, and numerical results are presented for problems on the two-sphere, the paraboloid and the Stiefel manifold.

PAPER 4: Dissipative numerical schemes on Riemannian manifolds with applications to gradient flows

by *Elena Celledoni, Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm*
Published in *SIAM Journal on Scientific Computing 40 (6), pp. A3789–A3806 (2018)*

Here we present the discrete Riemannian gradient method for dissipative systems on Riemannian manifolds, with an extra focus on gradient flows. Hence we employ a generalization of the Itoh–Abe discrete gradient, which has advantages over the other discrete gradients for such problems. Our scheme is demonstrated on eigenvalue problems and manifold valued image denoising problems, with implementation issues being discussed in detail.

PAPER 5: Order theory for discrete gradient methods

by *Sølve Eidnes*
Submitted

We present a general form for a class of the $\overline{S}(x,y,h)$ in (8) approximating $S(x)$, and conditions on this for reaching an arbitrary order of the corresponding discrete gradient method. We show how, by choosing the AVF discrete gradient, one obtains arbitrary order energy-preserving B-series methods for skew-gradient systems with constant S, and arbitrary order energy-preserving P-series methods for general skew-gradient systems.

PAPER 6: Linearly implicit structure-preserving schemes for Hamiltonian systems

by *Sølve Eidnes, Lu Li and Shun Sato*
To appear in *Journal of Computational and Applied Mathematics*

Despite a growing interest in both energy preservation and linearly implicit schemes for Hamiltonian PDEs over the last few decades, few studies have been performed on using Kahan's method for the time-stepping of PDEs with a cubic Hamiltonian. Here we compare Kahan's method to a linearly implicit generalization of discrete gradient methods, and test these methods on the KdV and Camassa–Holm equations. The numerical results and analysis of the methods point towards Kahan's method being the favorable choice.

PAPER 7: Linearly implicit local and global energy-preserving methods

by *Sølve Eidnes and Lu Li*
Submitted

Hamiltonian PDEs with a multi-symplectic structure have three local conservation laws. We show that we can preserve discrete approximations to the local energy conservation laws by applying Kahan's method for the temporal integration, if the energy is cubic. Numerical examples are performed for the KdV equation and the two-dimensional Zakharov–Kuznetsov equation, yielding beneficial results compared to fully implicit schemes.

PAPER 8: Shape analysis on Lie groups and homogeneous spaces

by *Elena Celledoni, Sølve Eidnes, Markus Eslitzbichler and Alexander Schmeding*
Published in *Proceedings for Geometric Science of Information 2017, Lecture Notes in Computer Science 10589, pp. 49–56 (2017)*

This conference proceedings presents an approach to shape analysis built upon the square root velocity transform (SRVT) generalised to Lie groups and homogeneous spaces. It presents the main ideas behind the methods developed in [10] and the later Paper 9, without delving into the details.

PAPER 9: Shape analysis on homogeneous spaces: a generalised SRVT framework

by *Elena Celledoni, Sølve Eidnes and Alexander Schmeding*
Published in *Abel Symposia 13, pp. 187–220 (2018)*

Here we present in detail a generalised SRVT framework for shape analysis on homogeneous spaces, using Lie group actions. Different Lie group actions lead to different metrics, opening up for a variety of possibilies which we show can be implemented in the same framework. We demonstrate our method by applying it to the matching of two curves on the two-sphere.

Bibliography

[1] I. Babuška and B. Guo. The h, p and h-p version of the finite element method; basis theory and applications. *Adv. Eng. Softw.*, 15:159–174, 1992.

[2] M. Bauer, M. Bruveris, S. Marsland, and P. W. Michor. Constructing reparameterization invariant metrics on spaces of plane curves. *Differential Geom. Appl.*, 34:139–165, 2014.

[3] S. Blanes and F. Casas. *A concise introduction to geometric numerical integration*. Monographs and Research Notes in Mathematics. CRC Press, Boca Raton, FL, 2016.

[4] 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(1962):1365–1395, 1997.

[5] T. J. Bridges. Multi-symplectic structures and wave propagation. *Math. Proc. Cambridge Philos. Soc.*, 121(1):147–190, 1997.

[6] T. J. Bridges and S. Reich. Multi-symplectic integrators: numerical schemes for Hamiltonian PDEs that conserve symplecticity. *Phys. Lett. A*, 284(4-5):184–193, 2001.

[7] C. J. Budd, W. Huang, and R. D. Russell. Moving mesh methods for problems with blow-up. *SIAM J. Sci. Comput.*, 17(2):305–327, 1996.

[8] C. J. Budd, W. Huang, and R. D. Russell. Adaptivity with moving grids. *Acta Numer.*, 18:111–241, 2009.

[9] W. Cai, H. Li, and Y. Wang. Partitioned averaged vector field methods. *J. Comput. Phys.*, 370:25–42, 2018.

[10] E. Celledoni, M. Eslitzbichler, and A. Schmeding. Shape analysis on Lie groups with applications in computer animation. *J. Geom. Mech.*, 8(3):273–304, 2016.

[11] 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(20):6770–6789, 2012.

[12] E. Celledoni, H. Marthinsen, and B. Owren. An introduction to Lie group integrators – basics, new developments and applications. *J. Comput. Phys.*, 257:1040–1061, 2014.

[13] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, and G. R. W. Quispel. Integrability properties of Kahan's method. *J. Phys. A*, 47(36):365202, 20, 2014.

[14] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, G. R. W. Quispel, and W. M. Wright. Energy-preserving Runge-Kutta methods. *M2AN Math. Model. Numer. Anal.*, 43(4):645–649, 2009.

[15] E. Celledoni, R. I. McLachlan, B. Owren, and G. R. W. Quispel. Geometric properties of Kahan's method. *J. Phys. A*, 46(2):025201, 12, 2013.

[16] E. Celledoni, D. I. McLaren, B. Owren, and G. R. W. Quispel. Geometric and integrability properties of Kahan's method: the preservation of certain quadratic integrals. *J. Phys. A*, 52(6):065201, 9, 2019.

[17] E. Celledoni and B. Owren. Preserving first integrals with symmetric Lie group methods. *Discrete Contin. Dyn. Syst.*, 34(3):977–990, 2014.

[18] A. J. Chorin, M. F. McCracken, T. J. R. Hughes, and J. E. Marsden. Product formulas and numerical algorithms. *Comm. Pure Appl. Math.*, 31(2):205–256, 1978.

[19] D. Cohen and E. Hairer. Linear energy-preserving integrators for Poisson systems. *BIT*, 51(1):91–101, 2011.

[20] G. J. Cooper. Stability of Runge-Kutta methods for trajectory problems. *IMA J. Numer. Anal.*, 7(1):1–13, 1987.

[21] P. E. Crouch and R. Grossman. Numerical integration of ordinary differential equations on manifolds. *J. Nonlinear Sci.*, 3(1):1–33, 1993.

[22] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33(5):2318–2340, 2011.

[23] M. J. Ehrhardt, E. S. Riis, T. Ringholm, and C.-B. Schönlieb. A geometric integration approach to smooth optimisation: Foundations of the discrete gradient method. *arXiv preprint, arXiv:1805.06444*, 2018.

[24] K. Feng and M. Z. Qin. The symplectic methods for the computation of Hamiltonian equations. In *Numerical methods for partial differential equations (Shanghai, 1987)*, volume 1297 of *Lecture Notes in Math.*, pages 1–37. Springer, Berlin, 1987.

[25] D. Furihata. Finite difference schemes for $\partial u/\partial t = (\partial/\partial x)^\alpha \delta G/\delta u$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156(1):181–205, 1999.

[26] D. Furihata and T. Matsuo. *Discrete variational derivative method*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011. A structure-preserving numerical method for partial differential equations.

[27] Y. Gong, J. Cai, and Y. Wang. Some new structure-preserving algorithms for general multi-symplectic formulations of Hamiltonian PDEs. *J. Comput. Phys.*, 279:80–102, 2014.

[28] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[29] V. Grimm, R. I. McLachlan, D. I. McLaren, G. R. W. Quispel, and C.-B. Schönlieb. Discrete gradient methods for solving variational image regularisation models. *J. Phys. A*, 50(29):295201, 21, 2017.

[30] E. Hairer. Energy-preserving variant of collocation methods. *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, 5(1-2):73–84, 2010.

[31] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[32] E. Hairer and G. Wanner. *Solving ordinary differential equations. II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1996. Stiff and differential-algebraic problems.

[33] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, 25(1):35–61, 1983.

[34] W. Huang and R. Russell. *Adaptive moving mesh methods*, volume 174 of *Springer Series in Applied Mathematical Sciences*. Springer-Verlag, New York, 2010.

[35] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numer.*, 9:215–365, 2000.

[36] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 76(1):85–102, 1988.

[37] C. Jiang, Y. Gong, W. Cai, and Y. Wang. A linearly implicit structure-preserving scheme for the Camassa–Holm equation based on multiple scalar auxiliary variables approach. *arXiv preprint, arXiv:1907.00167*, 2019.

[38] W. Kahan. Unconventional numerical methods for trajectory calculations. *Unpublished lecture notes*, 1:13, 1993.

[39] W. Kahan and R.-C. Li. Unconventional schemes for a class of ordinary differential equations—with applications to the Korteweg-de Vries equation. *J. Comput. Phys.*, 134(2):316–331, 1997.

[40] Y. A. Kriksin. A conservative difference scheme for a system of Hamiltonian equations with external action. *Zh. Vychisl. Mat. i Mat. Fiz.*, 33(2):206–218, 1993.

[41] S. Kurtek, A. Srivastava, E. Klassen, and Z. Ding. Statistical modeling of curves using shapes and related features. *J. Amer. Statist. Assoc.*, 107(499):1152–1165, 2012.

[42] T. Lee, M. Baines, and S. Langdon. A finite difference moving mesh method based on conservation for moving boundary problems. *J. Comput. Appl. Math.*, 288:1–17, 2015.

[43] B. Leimkuhler and G. W. Patrick. A symplectic integrator for Riemannian manifolds. *J. Nonlinear Sci.*, 6(4):367–384, 1996.

[44] B. Leimkuhler and S. Reich. *Simulating Hamiltonian dynamics*, volume 14 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2004.

[45] J. E. Marsden. *Lectures on mechanics*, volume 174 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 1992.

[46] J. E. Marsden, G. W. Patrick, and S. Shkoller. Multisymplectic geometry, variational integrators, and nonlinear PDEs. *Comm. Math. Phys.*, 199(2):351–395, 1998.

[47] T. Matsuo. New conservative schemes with discrete variational derivatives for nonlinear wave equations. *J. Comput. Appl. Math.*, 203(1):32–56, 2007.

[48] T. Matsuo and D. Furihata. Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations. *J. Comput. Phys.*, 171(2):425–447, 2001.

[49] T. Matsuo, M. Sugihara, D. Furihata, and M. Mori. Spatially accurate dissipative or conservative finite difference schemes derived by the discrete variational method. *Japan J. Indust. Appl. Math.*, 19(3):311–330, 2002.

[50] T. Matsuo, M. Sugihara, D. Furihata, and M. Mori. Spatially accurate dissipative or conservative finite difference schemes derived by the discrete variational method. *Japan J. Indust. Appl. Math.*, 19(3):311–330, 2002.

[51] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci.*, 357(1754):1021–1045, 1999.

[52] R. I. McLachlan, G. R. W. Quispel, and G. S. Turner. Numerical integrators that preserve symmetries and reversing symmetries. *SIAM J. Numer. Anal.*, 35(2):586–599, 1998.

[53] D. I. McLaren and G. R. W. Quispel. Integral-preserving integrators. *J. Phys. A*, 37(39):L489–L495, 2004.

[54] D. I. McLaren and G. R. W. Quispel. Bootstrapping discrete-gradient integral-preserving integrators to fourth order. In M. Daniel and S. Rajasekar, editors, *Nonlinear dynamics*, pages 157–171. Narosa Publishing House, 2009.

[55] P. W. Michor and D. Mumford. Riemannian geometries on spaces of plane curves. *J. Eur. Math. Soc. (JEMS)*, 8(1):1–48, 2006.

[56] Y. Miyatake and T. Matsuo. A note on the adaptive conservative/dissipative discretization for evolutionary partial differential equations. *J. Comput. Appl. Math.*, 274:79–87, 2015.

[57] H. Munthe-Kaas. Lie–Butcher theory for Runge–Kutta methods. *BIT*, 35(4):572–587, 1995.

[58] H. Munthe-Kaas. Runge–Kutta methods on Lie groups. *BIT*, 38(1):92–111, 1998.

[59] R. A. Norton, D. I. McLaren, G. R. W. Quispel, A. Stern, and A. Zanna. Projection methods and discrete gradient methods for preserving first integrals of ODEs. *Discrete Contin. Dyn. Syst.*, 35(5):2079–2098, 2015.

[60] R. A. Norton and G. R. W. Quispel. Discrete gradient methods for preserving a first integral of an ordinary differential equation. *Discrete Contin. Dyn. Syst.*, 34(3):1147–1170, 2014.

[61] P. J. Olver. *Applications of Lie groups to differential equations*, volume 107 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 1993.

[62] G. R. W. Quispel. Volume-preserving integrators. *Phys. Lett. A*, 206(1-2):26–30, 1995.

[63] G. R. W. Quispel and D. I. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A*, 41(4):045206, 7, 2008.

[64] G. R. W. Quispel and G. S. Turner. Discrete gradient methods for solving ODEs numerically while preserving a first integral. *J. Phys. A*, 29(13):L341–L349, 1996.

[65] E. S. Riis, M. J. Ehrhardt, G. Quispel, and C.-B. Schönlieb. A geometric integration approach to nonsmooth, nonconvex optimisation. *arXiv preprint, arXiv:1807.07554*, 2018.

[66] T. Ringholm, J. Lazić, and C.-B. Schönlieb. Variational image regularization with Euler's elastica using a discrete gradient scheme. *SIAM J. Imaging Sci.*, 11(4):2665–2691, 2018.

[67] J. M. Sanz-Serna and M. P. Calvo. *Numerical Hamiltonian problems*, volume 7 of *Applied Mathematics and Mathematical Computation*. Chapman & Hall, London, 1994.

[68] T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, 2003.

[69] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn. Shape analysis of elastic curves in Euclidean spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1415–1428, 2010.

[70] D. Wang, A. Xiao, and W. Yang. A linearly implicit conservative difference scheme for the space fractional coupled nonlinear Schrödinger equations. *J. Comput. Phys.*, 272:644–655, 2014.

[71] T. Yaguchi, T. Matsuo, and M. Sugihara. An extension of the discrete variational method to nonuniform grids. *J. Comput. Phys.*, 229(11):4382–4423, 2010.

[72] T. Yaguchi, T. Matsuo, and M. Sugihara. The discrete variational derivative method based on discrete differential forms. *J. Comput. Phys.*, 231(10):3963–3986, 2012.

[73] T. Yaguchi, T. Matsuo, and M. Sugihara. The discrete variational derivative method based on discrete differential forms. *J. Comput. Phys.*, 231(10):3963–3986, 2012.

[74] X. Yang, J. Zhao, and Q. Wang. Numerical approximations for the molecular beam epitaxial growth model based on the invariant energy quadratization method. *J. Comput. Phys.*, 333:104–127, 2017.

[75] P. A. Zegeling. *r*-refinement for evolutionary PDEs with finite elements or finite differences. *Appl. Numer. Math.*, 26:97–104, 1998.

Adaptive energy preserving methods for partial differential equations

Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm

Published in Advances in Computational Mathematics 44 (3), pp. 815–839 (2018)

Adaptive energy preserving methods for partial differential equations

Abstract. A framework for constructing integral preserving numerical schemes for time-dependent partial differential equations on non-uniform grids is presented. The approach can be used with both finite difference and partition of unity methods, thereby including finite element methods. The schemes are then extended to accommodate r-, h- and p-adaptivity. To illustrate the ideas, the method is applied to the Korteweg–de Vries equation and the sine-Gordon equation. Results from numerical experiments are presented.

1.1 Introduction

Difference schemes with conservation were introduced by Courant, Friedrichs and Lewy in [8], where a discrete conservation law for a finite difference approximation of the wave equation was derived. Their methods are often called energy methods [11] or energy-conserving methods [18], although the conserved quantity is often not energy in the physical sense. The primary motivation for developing conservative methods was originally to devise a norm that could guarantee global stability. This was still an objective, in addition to proving existence and uniqueness of solutions, when the energy methods garnered newfound interest in the 1950s and 1960s, resulting in new developments such as generalizations of the methods and more difference schemes, summarized by Richtmyer and Morton in [26].

In the 1970s, the motivation behind studying schemes that preserve invariant quantities changed, as the focus shifted to the conservation property itself. Li and Vu-Quoc presented in [18] a historical survey of conservative methods developed up to the early 1990s. They state that this line of work is motivated by the fact that in some situations, the success of a numerical solution will depend on its ability to preserve one or more of the invariant properties of the original differential equation. In addition, as noted in [7,14], there is the general idea that transferring more of the properties of the original continuous dynamical system over to a discrete dynamical system may lead to a more accurate numerical approximation of the solution, especially over long time intervals.

In recent years, there has been a greater interest in developing systematic techniques applicable to larger classes of differential equations. Hairer, Lubich and Wanner give in [14] a presentation of geometric integrators for differential equations, i.e. methods for solving ordinary differential equations (ODEs) that preserve a geometric structure of the system. Examples of such geometric structures are symplectic structures, symmetries, reversing symmetries, isospectrality, Lie group structure, orthonormality, first integrals, and other invariants, such as volume and invariant measure.

In this paper we will be concerned with the preservation of first integrals of PDEs. From the ODE literature we find that the most general methods for preserving first integrals are tailored schemes, in the sense that the vector field of the ODE does not by itself provide sufficient information, so the schemes make explicit use of the first integral. An obvious approach in this respect is projection, where the solution is first advanced using any consistent numerical scheme and then this approximation is projected onto the appropriate level set of the invariant. In the same class of tailored methods one also has the discrete gradient methods, usually attributed to Gonzalez [13]. For the subclass of canonical Hamiltonian systems, the energy can be preserved by means of a general purpose method called the averaged vector field method, see e.g. [25].

The notion of discrete gradient methods for ordinary differential equations has a counterpart for partial differential equations called the discrete variational derivative method. Such schemes have been developed since the late 1990s in a number of articles by Japanese researchers such as Furihata, Matsuo, Sugihara, and Yaguchi. A relatively recent account of this work can be found in the monograph [12]. More recently, the development of integral preserving schemes for PDEs has been systematised and eased, in particular by using the aforementioned tools from ordinary differential equations, see for instance [6,9]. Most of the schemes one finds in the literature are based on a finite difference approach, and usually on fixed, uniform grids. There are however some exceptions. Yaguchi, Matsuo and Sugihara presented in [27,28] two different discrete variational derivative methods on fixed, non-uniform grids, specifically defined for certain classes of PDEs. Non-uniform grids are of particular importance for multidimensional problems, since the use of uniform grids will greatly restrict the types of domains possible to discretize. Another important consequence of being able to use non-uniform grids is that it allows for the use of time-adaptive spatial meshes for solving partial differential equations. Adaptive energy preserving schemes for the Korteweg–de Vries and Cahn–Hilliard equations have been developed recently [22] by Miyatake and Matsuo. The main objective of this paper is to propose a general framework for numerical methods for PDEs that combine mesh adaptivity with first integral conservation.

Several forms of adaptive methods exist, and they can roughly be categorized as r-, h- and p-adaptive. When applying r-adaptivity, one keeps the number of degrees of freedom constant while modifying the mesh at each time step to e.g. cluster in problematic areas such as boundary layers or to follow wave fronts. When applying the Finite Difference Method (FDM) or the Finite Element Method (FEM), moving mesh methods may be used for r-adaptivity, some examples of which may be found in [16, 17, 29]. When using Partition of Unity Methods (PUM) (and in particular when using FEM), h- and p-adaptivity relate to adjusting the number of elements and the basis functions used on the elements, respectively. For PUM methods there exist strategies for h- and p-adaptivity based both on a priori and a posteriori error analysis [1]. Common to all of these strategies is that, based on estimated function values in preceding time steps, one can suggest improved discretization parameters for the next time step. In the FDM approach, these discretization parameters consist of the mesh points $\mathbf{x}$, while in the PUM approach the parameters encompass information about both the mesh and the basis functions. We will, in general, denote a collection of discretization parameters by $\mathbf{p}$, and assume that the discretization parameters are changed separately from the degrees of freedom $\mathbf{u}$ of the problem when using adaptive methods. That is, starting with an initial set of discretization parameters $\mathbf{p}^0$ and initial values $\mathbf{u}^0$, one first decides upon $\mathbf{p}^1$ before calculating $\mathbf{u}^1$, then finding $\mathbf{p}^2$, then $\mathbf{u}^2$, etc., in a decoupled fashion.

A first integral of a PDE is a functional $\mathcal{I}$ on an infinite-dimensional space, yet our numerical methods will reduce the problem to a finite-dimensional setting. Therefore, we cannot preserve the exact value of the first integral; instead, we will preserve a consistent approximation to the first integral, $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$. The approximation will be dependent on the discretization parameters $\mathbf{p}$ and, since adaptivity alters the values of $\mathbf{p}$, we will therefore aim to preserve the value of the approximated first integral across all discretization parameters, i.e. we will require that $\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) = \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)$. Here, and in the following, superscripts denote time steps unless otherwise specified.

In this article, we present a method for developing adaptive numerical schemes that conserve an approximated first integral. In Section 2, the PDE problem is stated, and two classes of first integral preserving methods using arbitrary, constant discretization parameters are presented; one using an FDM approach and the other a PUM approach for spatial discretization. A connection to existing methods is then established. In Section 3, we present a way of adding adaptivity to the methods from Section 2 and the modifications needed to retain the first integral preservation property, before showing that certain projection methods form a subclass of the methods thus obtained. Section 4 contains examples of the methods applied to two PDEs and numerical results assessing the quality of the numerical solutions as compared to a standard implicit method.

1.2 Spatial discretization with fixed mesh

1.2.1 Problem statement

Consider a partial differential equation

$$u_t = f(\mathbf{x}, u^J), \qquad \mathbf{x} \in \Omega \subseteq \mathbb{R}^d, \quad u \in \mathcal{B} \subseteq L^2, \tag{1.1}$$

where u^J denotes u itself and its partial derivatives of any order with respect to the spatial variables $x_1, ..., x_d$. We shall not specify the space $\mathcal{B}$ further, but assume that it is sufficiently regular to allow all operations used in the following. For ease of reading, all t-dependence will be suppressed in the notation wherever it is irrelevant. Also, from here on, square brackets are used to denote dependence on a function and its partial derivatives of any order with respect to the independent variables t and $x_1, ..., x_d$. We recall the definition of the *variational derivative* of a functional $H[u]$ as the function $\frac{\delta H}{\delta u}[u]$ satisfying

$$\left\langle \frac{\delta H}{\delta u}[u], v \right\rangle_{L^2} = \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} H[u+\epsilon v] \quad \forall v \in \mathcal{B}, \tag{1.2}$$

and define a *first integral* of (1.1) to be a functional $\mathcal{I}[u]$ satisfying

$$\left\langle \frac{\delta \mathcal{I}}{\delta u}[u], f(\mathbf{x}, u^J) \right\rangle_{L^2} = 0, \quad \forall u \in \mathcal{B}.$$

We may observe that $\mathcal{I}[u]$ is preserved over time, since this implies

$$\frac{\mathrm{d}\mathcal{I}}{\mathrm{d}t} = \left\langle \frac{\delta \mathcal{I}}{\delta u}[u], \frac{\partial u}{\partial t} \right\rangle_{L^2} = 0.$$

Furthermore, we may observe that if there exists an operator $S(\mathbf{x}, u^J)$, skew-symmetric with respect to the L^2 inner product, such that

$$f(\mathbf{x}, u^J) = S(\mathbf{x}, u^J)\frac{\delta \mathcal{I}}{\delta u}[u],$$

then $\mathcal{I}[u]$ is a first integral of (1.1), and we can state (1.1) in the form

$$u_t = S(\mathbf{x}, u^J)\frac{\delta \mathcal{I}}{\delta u}[u]. \tag{1.3}$$

This can be considered as the PDE analogue of an ODE with a first integral, in which case we have a system

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = S(\mathbf{u})\nabla_{\mathbf{u}} I(\mathbf{u}), \tag{1.4}$$

where $S(\mathbf{u})$ is a skew-symmetric matrix [20]. The gradient is defined as usual, but for clarity in later use we have added a subscript to specify that it is a vector of partial derivatives with respect to the coordinates of $\mathbf{u}$. Note that Hamiltonian equations are contained of this class of ODEs. For such differential equations, there exist numerical methods preserving the first integral $I(\mathbf{u})$, for instance the discrete gradient methods, which are of the form

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \bar{S}(\mathbf{u}^n, \mathbf{u}^{n+1})\overline{\nabla} I(\mathbf{u}^n, \mathbf{u}^{n+1}),$$

where $\bar{S}(\mathbf{u}^n, \mathbf{u}^{n+1})$ is a consistent skew-symmetric time-discrete approximation to $S(\mathbf{u})$ and $\overline{\nabla} I(\mathbf{v}, \mathbf{u})$ is a discrete gradient of $I(\mathbf{u})$, i.e. a function satisfying

$$(\overline{\nabla} I(\mathbf{v}, \mathbf{u}))^T (\mathbf{u} - \mathbf{v}) = I(\mathbf{u}) - I(\mathbf{v}), \tag{1.5}$$

$$\overline{\nabla} I(\mathbf{u}, \mathbf{u}) = \nabla_{\mathbf{u}} I(\mathbf{u}). \tag{1.6}$$

There are several possible choices of discrete gradients available, one of which is the Average Vector Field (AVF) discrete gradient [6], given by

$$\overline{\nabla} I(\mathbf{v}, \mathbf{u}) = \int_0^1 \nabla_{\mathbf{u}} I(\xi \mathbf{u} + (1 - \xi)\mathbf{v}) \mathrm{d}\xi,$$

which will be used for numerical experiments in the final chapter. Our approach to solving (1.1) on non-uniform grids is based upon considering the PDE in the form (1.3), reducing it to a system of ODEs of the form (1.4) and applying a discrete gradient method. This is done by finding a discrete approximation $\mathcal{I}_{\mathbf{p}}$ to $\mathcal{I}$ and using this to obtain a discretization in the spatial variables, which is achieved through either a finite difference approach or a variational approach.

1.2.2 Finite difference method

In the finite difference approach, we restrict ourselves to obtaining approximate values of u at the grid points $\mathbf{x}_0, ..., \mathbf{x}_M$, which can be interpreted as quadrature points with some associated nonzero quadrature weights $\kappa_0, ..., \kappa_M$. The grid points constitute the discretization parameters $\mathbf{p}$. We can then approximate the L^2 inner product by quadrature to arrive at a weighted inner product:

$$\langle u, v \rangle_{L^2} = \int_\Omega u(\mathbf{x}) v(\mathbf{x}) \mathrm{d}x \simeq \sum_{i=0}^{M} \kappa_i u(\mathbf{x}_i) v(\mathbf{x}_i) = \mathbf{u}^T D(\kappa) \mathbf{v} = \langle \mathbf{u}, \mathbf{v} \rangle_\kappa,$$

where $D(\kappa) = \mathrm{diag}(\kappa_0, ..., \kappa_M)$. Assume that there exists a consistent approximation $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$ to the functional $\mathcal{I}[u]$, dependent on the values of u at the points $\mathbf{x}_i$. Then, we can characterize the discretized variational derivative by asserting that

$$\left\langle \frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta \mathbf{u}}(\mathbf{u}), \mathbf{v} \right\rangle_\kappa = \left. \frac{\mathrm{d}}{\mathrm{d}\epsilon} \right|_{\epsilon=0} \mathcal{I}_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v}) \quad \forall \mathbf{v} \in \mathbb{R}^{M+1},$$

meaning

$$\left(\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta \mathbf{u}}(\mathbf{u}) \right)^T D(\kappa) \mathbf{v} = (\nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\mathbf{u}))^T \mathbf{v} \quad \forall \mathbf{v} \in \mathbb{R}^{M+1},$$

from which we conclude that

$$\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta \mathbf{u}}(\mathbf{u}) = D(\kappa)^{-1} \nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\mathbf{u}). \tag{1.7}$$

Using this as a discretization of $\frac{\delta \mathcal{I}}{\delta u}[u]$ and approximating $S(\mathbf{x}, u^J)$ by a matrix $S_d(\mathbf{u})$, skew-symmetric with respect to $\langle \cdot, \cdot \rangle_\kappa$, we obtain a discretization of (1.3) as:

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = S_{\mathbf{p}}(\mathbf{u}) \nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\mathbf{u}), \tag{1.8}$$

where $S_{\mathbf{p}}(\mathbf{u}) = S_d(\mathbf{u}) D(\kappa)^{-1}$. This system of ODEs is of the form (1.4), since

$$\begin{aligned} S_{\mathbf{p}}(\mathbf{u})^T &= (S_d(\mathbf{u}) D(\kappa)^{-1})^T \\ &= D(\kappa)^{-1} S_d(\mathbf{u})^T D(\kappa) D(\kappa)^{-1} \\ &= -D(\kappa)^{-1} D(\kappa) S_d(\mathbf{u}) D(\kappa)^{-1} \\ &= -S_d(\mathbf{u}) D(\kappa)^{-1} \\ &= -S_{\mathbf{p}}(\mathbf{u}). \end{aligned}$$

This allows us to apply first integral preserving methods for systems of ODEs to solve the spatially discretized system. For example, we may consider using a discrete gradient $\overline{\nabla} \mathcal{I}_{\mathbf{p}}$, and a skew-symmetric, time-discrete approximation $S_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1})$ to $S_{\mathbf{p}}(\mathbf{u})$, where $\mathbf{u}^n = \mathbf{u}(t_n)$, $t_n = n \Delta t$. Then, the following scheme will preserve the approximated first integral $\mathcal{I}_{\mathbf{p}}$ in the sense that $\mathcal{I}_{\mathbf{p}}(\mathbf{u}^{n+1}) = \mathcal{I}_{\mathbf{p}}(\mathbf{u}^n)$:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = S_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}) \overline{\nabla} \mathcal{I}_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}). \tag{1.9}$$

1.2.3 Partition of unity method

One may also approach the problem of spatially discretizing the PDE through the use of variational methods such as the Partition of Unity Method (PUM) [21], which generalizes the Finite Element Method (FEM). Here, the variational structure of the functional derivative can be utilized in a natural way, such that one avoids having to approximate $S(\mathbf{x}, u^J)$. We begin by stating a weak form of (1.3). Then, the problem consists of finding $u \in \mathcal{B}$ such that

$$\langle u_t, v \rangle_{L^2} = \left\langle S(\mathbf{x}, u^J) \frac{\delta \mathcal{I}}{\delta u}[u], v \right\rangle_{L^2} = -\left\langle \frac{\delta \mathcal{I}}{\delta u}[u], S(\mathbf{x}, u^J) v \right\rangle_{L^2} \quad \forall v \in \mathcal{B}. \tag{1.10}$$

Employing a Galerkin formulation, we restrict the search to a finite dimensional subspace $\mathcal{B}^h = \mathrm{span}\{\varphi_0, ...\varphi_M\} \subseteq \mathcal{B}$, and approximate u by the function

$$u^h(x,t) = \sum_{i=0}^{M} u_i(t)\varphi_i(x).$$

We denote by $\mathbf{p}$ the collection of discretization parameters defining $\mathcal{B}^h$; this includes information about mesh points, element types and shapes of basis functions. Furthermore, we define the canonical mapping $\Phi_{\mathbf{p}} : \mathbb{R}^{M+1} \to \mathcal{B}^h$ given by

$$\Phi_{\mathbf{p}}(\mathbf{u}) = \sum_{i=0}^{M} u_i \varphi_i, \tag{1.11}$$

and the discrete first integral $\mathcal{I}_{\mathbf{p}}$ by

$$\mathcal{I}_{\mathbf{p}}(\mathbf{u}) = \mathcal{I}(\Phi_{\mathbf{p}}(\mathbf{u})).$$

The following lemma will prove useful later in the construction of the method:

Lemma 1.1. *For any $u^h, v \in \mathcal{B}^h$,*

$$\left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \mathcal{I}(u^h + \epsilon v) = (\nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\mathbf{u}))^T \mathbf{v}.$$

Proof.

$$\begin{aligned}
\left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \mathcal{I}(u^h + \epsilon v) &= \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \mathcal{I}(\Phi_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v})) \\
&= \left.\left\langle \frac{\delta \mathcal{I}}{\delta u}[\Phi_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v})], \frac{\mathrm{d}}{\mathrm{d}\epsilon}\Phi_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v}) \right\rangle_{L^2}\right|_{\epsilon=0} \\
&= \left.\left\langle \frac{\delta \mathcal{I}}{\delta u}[\Phi_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v})], (\nabla_{\mathbf{u}}\Phi_{\mathbf{p}}(\mathbf{u} + \epsilon \mathbf{v}))^T \mathbf{v} \right\rangle_{L^2}\right|_{\epsilon=0} \\
&= \left\langle \frac{\delta \mathcal{I}}{\delta u}[\Phi_{\mathbf{p}}(\mathbf{u})], (\nabla_{\mathbf{u}}\Phi_{\mathbf{p}}(\mathbf{u}))^T \mathbf{v} \right\rangle_{L^2} \\
&= \sum_{i=0}^{M} v_i \left\langle \frac{\delta \mathcal{I}}{\delta u}[\Phi_{\mathbf{p}}(\mathbf{u})], \frac{\partial}{\partial u_i}\Phi_{\mathbf{p}}(\mathbf{u}) \right\rangle_{L^2} \\
&= \sum_{i=0}^{M} v_i \frac{\partial}{\partial u_i}\mathcal{I}[\Phi_{\mathbf{p}}(\mathbf{u})] = \sum_{i=0}^{M} v_i \frac{\partial}{\partial u_i}\mathcal{I}_{\mathbf{p}}(\mathbf{u}) = (\nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\mathbf{u}))^T \mathbf{v}.
\end{aligned}$$

□

We observe that for $u, v \in \mathcal{B}^h$, the L^2 inner product has a discrete counterpart:

$$\langle u, v\rangle_{L^2} = \sum_{i=0}^{M}\sum_{j=0}^{M} u_i v_j \left\langle \varphi_i, \varphi_j \right\rangle_{L^2} = \mathbf{u}^T A\mathbf{v} = \langle \mathbf{u}, \mathbf{v}\rangle_A$$

with the symmetric positive definite matrix A given by $A_{ij} = \left\langle \varphi_i, \varphi_j \right\rangle_{L^2}$. Note also that equation (1.10) is satisfied in $\mathcal{B}^h$ if it is satisfied for all basis functions φ_j. The Galerkin form of the problem therefore consists of finding $u_i(t)$ such that

$$\sum_{i=0}^{M} \frac{\mathrm{d}u_i}{\mathrm{d}t} \left\langle \varphi_i, \varphi_j \right\rangle_{L^2} = -\left\langle \frac{\delta \mathcal{I}}{\delta u}[u^h], S(\mathbf{x}, u^{h,J})\varphi_j \right\rangle_{L^2} \quad \forall j \in \{0, ..., M\}. \tag{1.12}$$

This weak form is rather unwieldy and does not give rise to a system of the form (1.4), so in order to make further progress, we consider the projection of $\frac{\delta \mathcal{I}}{\delta u}[u^h]$ onto $\mathcal{B}^h$:

$$\frac{\delta \mathcal{I}^h}{\delta u}[u^h] = \sum_{i=0}^{M} w_i^h[u^h]\varphi_i(x) = \sum_{i=0}^{M} w_i(\mathbf{u})\varphi_i(x),$$

where $w_i(\mathbf{u}) = w_i^h[\Phi(\mathbf{u})] = w_i^h[u^h]$ are coefficients that will be characterized later. Replacing $\frac{\delta \mathcal{I}}{\delta u}[u^h]$ by its projection in (1.12) gives the approximate weak form:

$$\sum_{i=0}^{M} \frac{\mathrm{d}u_i}{\mathrm{d}t} \left\langle \varphi_i, \varphi_j \right\rangle_{L^2} = -\sum_{i=0}^{M} w_i(\mathbf{u}) \left\langle \varphi_i, S(\mathbf{x}, u^{h,J})\varphi_j \right\rangle_{L^2} \quad \forall j \in \{0, ..., M\}.$$

Thus, we obtain a system of equations for the coefficients u_i:

$$A\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -B(\mathbf{u})\mathbf{w}(\mathbf{u}), \tag{1.13}$$

with the skew-symmetric matrix $B(\mathbf{u})$ given by $B(\mathbf{u})_{ji} = \left\langle \varphi_i, S(\mathbf{x}, \Phi(\mathbf{u})^J)\varphi_j \right\rangle_{L^2}$. Furthermore, we may characterize the vector $\mathbf{w}(\mathbf{u})$ by the following argument:

$$\begin{aligned}\mathbf{w}(\mathbf{u})^T A\mathbf{v} &= \left\langle \frac{\delta \mathcal{I}^h}{\delta u}[u^h], v \right\rangle_{L^2} = \left\langle \frac{\delta \mathcal{I}}{\delta u}[u^h], v \right\rangle_{L^2} \\ &= \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \mathcal{I}(u^h + \epsilon v) = (\nabla_{\mathbf{u}}\mathcal{I}_{\mathbf{p}}(\mathbf{u}))^T\mathbf{v},\end{aligned}$$

where the last equality holds by Lemma 1.1. This holds for all $\mathbf{v} \in \mathbb{R}^{M+1}$, and thus

$$\mathbf{w}(\mathbf{u}) = A^{-1}\nabla_{\mathbf{u}}\mathcal{I}_{\mathbf{p}}(\mathbf{u}). \tag{1.14}$$

Inserting (1.14) into (1.13) and left-multiplying by A^{-1}, we are left with an ODE for the coefficients u_i:

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = S_{\mathbf{p}}(\mathbf{u})\nabla_{\mathbf{u}}\mathcal{I}_{\mathbf{p}}(\mathbf{u}). \tag{1.15}$$

Here, $S_{\mathbf{p}}(\mathbf{u}) = -A^{-1}B(\mathbf{u})A^{-1}$ is a skew-symmetric matrix, and the system is thereby of the form (1.4), meaning $\mathcal{I}_{\mathbf{p}}$ can be preserved numerically using e.g. discrete gradient methods as in equation (1.9).

1.2.4 Discrete variational derivative methods

Let us now define a general framework for the discrete variational derivative methods that encompass the methods presented by Furihata, Matsuo and coauthors in a number of publications including [10–12, 27, 28].

Definition 1.1. Let $\mathcal{I}_{\mathbf{p}}$ be a consistent approximation to the functional $\mathcal{I}[u]$ discretized on $\mathbf{p}$ given by grid points $\mathbf{x}_i$ and quadrature weights κ_i, $i = 0, ..., M$. Then $\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{v},\mathbf{u})}(\mathbf{v},\mathbf{u})$ is a discrete variational derivative of $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$ if it is a continuous function satisfying

$$\left\langle \frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{v},\mathbf{u})}, \mathbf{u}-\mathbf{v} \right\rangle_{\kappa} = \mathcal{I}_{\mathbf{p}}(\mathbf{u}) - \mathcal{I}_{\mathbf{p}}(\mathbf{v}), \tag{1.16}$$

$$\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{u},\mathbf{u})} = \frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta \mathbf{u}}(\mathbf{u}), \tag{1.17}$$

and the discrete variational derivative methods for solving PDEs on the form (1.3) are given by

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} = S_d(\mathbf{u}^n,\mathbf{u}^{n+1})\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{u}^n,\mathbf{u}^{n+1})}, \tag{1.18}$$

where $S_d(\mathbf{u}^n,\mathbf{u}^{n+1})$ is a time-discrete approximation to $S_d(\mathbf{u})$, skew-symmetric with respect to the inner product $\langle\cdot,\cdot\rangle_{\kappa}$.

Proposition 1.1. *A discrete gradient method (1.9) applied to the system of ODEs (1.8) or (1.15) is equivalent to a discrete variational derivative method as given by (1.18), with*

$$S_d(\mathbf{u}^n,\mathbf{u}^{n+1}) = S_{\mathbf{p}}(\mathbf{u}^n,\mathbf{u}^{n+1})D(\kappa),$$

and the discrete variational derivative

$$\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{v},\mathbf{u})} = D(\kappa)^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}}(\mathbf{v},\mathbf{u}) \tag{1.19}$$

satisfying (1.16)-(1.17).

Proof. Applying (1.5), we find, for the discrete variational derivative (1.19),

$$\begin{aligned}\left\langle \frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{v},\mathbf{u})}, \mathbf{u}-\mathbf{v} \right\rangle_{\kappa} &= \left\langle D(\kappa)^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}}(\mathbf{v},\mathbf{u}), \mathbf{u}-\mathbf{v} \right\rangle_{\kappa} \\ &= \left(D(\kappa)^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}}\left(\mathbf{v},\mathbf{u}\right)\right)^{\mathrm{T}} D(\kappa)\left(\mathbf{u}-\mathbf{v}\right) \\ &= \overline{\nabla}\mathcal{I}_{\mathbf{p}}\left(\mathbf{v},\mathbf{u}\right)^{\mathrm{T}}\left(\mathbf{u}-\mathbf{v}\right) = \mathcal{I}_{\mathbf{p}}(\mathbf{u}) - \mathcal{I}_{\mathbf{p}}(\mathbf{v}),\end{aligned}$$

and hence (1.16) is satisfied. Furthermore, applying (1.6) and (1.7),

$$\frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta(\mathbf{u},\mathbf{u})} = D(\kappa)^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}}(\mathbf{u},\mathbf{u}) = D(\kappa)^{-1}\nabla_{\mathbf{u}}\mathcal{I}_{\mathbf{p}}(\mathbf{u}) = \frac{\delta \mathcal{I}_{\mathbf{p}}}{\delta \mathbf{u}}(\mathbf{u})$$

and (1.17) is also satisfied. □

Hence, all discrete variational derivative methods as given by (1.18) can be expressed as discrete gradient methods on the system of ODEs (1.8) or (1.15) obtained by discretizing (1.3) in space, and vice versa.

1.3 Adaptive discretization

1.3.1 Mapping solutions between parameter sets

Assuming that adaptive strategies are employed, one would obtain a new set of discretization parameters $\mathbf{p}$ at each time step. After such a $\mathbf{p}$ has been found, the solution using the previous parameters must be transferred to the new parameter set before advancing to the next time step. This transfer procedure can be done in either a preserving or a non-preserving manner. Let $\mathbf{p}^n$, $\mathbf{u}^n$, $\mathbf{p}^{n+1}$ and $\mathbf{u}^{n+1}$ denote the discretization parameters and the numerical values obtained at the current time step and next time step, respectively. Also, let $\hat{\mathbf{u}}$ denote the values of $\mathbf{u}^n$ transferred onto $\mathbf{p}^{n+1}$ by whatever means. We call the transfer operation preserving if $\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) = \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)$. If the transfer is preserving, then the next time step can be taken with a preserving scheme, e.g.

$$\frac{\mathbf{u}^{n+1}-\hat{\mathbf{u}}}{\Delta t} = S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),$$

which is preserving in the sense that

$$\begin{aligned}\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) &= \mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) - \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) \\ &= \left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), \mathbf{u}^{n+1}-\hat{\mathbf{u}} \right\rangle \\ &= \Delta t \left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}) \right\rangle \\ &= 0,\end{aligned}$$

since $S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})$ is skew-symmetric. If non-preserving transfer is used, corrections are needed in order to obtain a preserving numerical method.

Proposition 1.2. *The scheme*

$$\mathbf{u}^{n+1} = \hat{\mathbf{u}} - \frac{(\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}))\mathbf{z}}{\left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),\mathbf{z}\right\rangle} + \Delta t S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), \quad (1.20)$$

where $\mathbf{z}$ *is an arbitrary vector chosen such that* $\left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),\mathbf{z}\right\rangle \neq 0$, *is first integral preserving in the sense that* $\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}) = 0$.

Proof.

$$\begin{aligned}
\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}) &= \mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) - \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) + \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}) \\
&= \left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), \mathbf{u}^{n+1} - \hat{\mathbf{u}} \right\rangle + \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}) \\
&= \left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), \mathbf{u}^{n+1} - \hat{\mathbf{u}} + \frac{(\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^{n}}(\mathbf{u}^{n}))\mathbf{z}}{\left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),\mathbf{z}\right\rangle} \right\rangle \\
&= \Delta t \left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}), S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}) \right\rangle \\
&= 0.
\end{aligned}$$

The second equality follows from (1.5), the fourth equality from the scheme (1.20), and the last equality follows from the skew-symmetry of $S_{\mathbf{p}^{n+1}}$. □

The correcting direction $\mathbf{z}$ should be chosen so as to obtain a minimal correction, and such that $\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),\mathbf{z}\rangle \neq 0$. One possibility is simply taking $\mathbf{z} = \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})$. In the FDM case one may alternatively choose $\mathbf{z} = D(\kappa)^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})$, and in the PUM case, $\mathbf{z} = A^{-1}\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})$.

When using the PUM formulation, one may obtain a method for preserving transfer in the following manner. Any changes through e.g. r- p- and/or h-refinement between time steps will result in a change in the shape and/or number of basis functions. Denote by $\mathcal{B}^h = \text{span}\{\varphi_i\}_{i=0}^{M}$ the trial space from the current time step and by $\hat{\mathcal{B}}^h = \text{span}\{\hat{\varphi}_i\}_{i=0}^{\hat{M}}$ the trial space for the next time step, and note that in general, $M \neq \hat{M}$. We do not concern ourselves with how the new basis is found, but simply acknowledge that the basis changes through adaptivity measures as presented in e.g. [16] or [1]. Our task is now to transfer the approximation u^h from $\mathcal{B}^h$ to $\hat{\mathcal{B}}^h$, obtaining an approximation $\hat{u}^h$, while conserving the first integral, i.e. $\mathcal{I}[u^h] = \mathcal{I}[\hat{u}^h]$. This can be formulated as a constrained minimization problem:

$$\min_{\hat{u}^h \in \tilde{\mathcal{B}}^{\langle}} ||\hat{u}^h - u^h||_{L^2}^2 \quad \text{s.t.} \quad \mathcal{I}[\hat{u}^h] = \mathcal{I}[u^h].$$

We observe that

$$\begin{aligned}||\hat{u}^h - u^h||_{L^2}^2 &= \sum_{i=0}^{\hat{M}} \sum_{j=0}^{\hat{M}} \hat{u}_i \hat{u}_j \hat{A}_{ij} - 2 \sum_{i=0}^{\hat{M}} \sum_{j=0}^{M} \hat{u}_i u_j^n C_{ij} + \sum_{i=0}^{M} \sum_{i=0}^{M} u_i^n u_j^n A_{ij} \\ &= \hat{\mathbf{u}}^T \hat{A} \hat{\mathbf{u}} - 2\hat{\mathbf{u}}^T C \mathbf{u}^n + \mathbf{u}^n A \mathbf{u}^n,\end{aligned}$$

where $A_{ij} = \langle \varphi_i, \varphi_j \rangle_{L^2}$, $\hat{A}_{ij} = \langle \hat{\varphi}_i, \hat{\varphi}_j \rangle_{L^2}$ and $C_{ij} = \langle \hat{\varphi}_i, \varphi_j \rangle_{L^2}$. Also observing that

$$\mathcal{I}[\hat{u}^h] = \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}), \quad \mathcal{I}[u^h] = \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n),$$

the problem can be reformulated as

$$\min_{\hat{\mathbf{u}} \in \mathbb{R}^{\hat{M}+1}} \hat{\mathbf{u}}^T \hat{A} \hat{\mathbf{u}} - 2\hat{\mathbf{u}}^T C \mathbf{u}^n + \mathbf{u}^n A \mathbf{u}^n \quad \text{s.t.} \quad \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) = 0.$$

This is a quadratic minimization problem with one nonlinear equality constraint. Using the method of Lagrange multipliers, we find $\hat{\mathbf{u}}$ as the solution of the nonlinear system of equations

$$\begin{aligned}\hat{A}\hat{\mathbf{u}} - C\mathbf{u}^n - \lambda \nabla_{\hat{\mathbf{u}}} \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) &= 0 \\ \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) &= 0,\end{aligned}$$

which can be solved numerically using a suitable nonlinear solver.

In general, applicable also in the FDM case, given $\bar{\mathbf{u}}$ obtained by interpolating $\mathbf{u}^n$ onto $\mathbf{p}^{n+1}$ in a non-preserving manner, a preserving transfer operation is obtained by solving the system of equations

$$\begin{aligned}\hat{\mathbf{u}} - \bar{\mathbf{u}} - \lambda \nabla_{\hat{\mathbf{u}}} \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) &= 0 \\ \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) &= 0.\end{aligned}$$

1.3.2 Projection methods

Let the function $f_{\mathbf{p}} : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}^M$ be such that

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = f_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}) \tag{1.21}$$

defines a step from time t_n to time t_{n+1} of any one-step method applied to (1.1) on the fixed grid represented by the discretization parameters $\mathbf{p}$. Then we define one step of an integral preserving linear projection method $\mathbf{u}^n \mapsto \mathbf{u}^{n+1}$ from $\mathbf{p}^n$ to $\mathbf{p}^{n+1}$ by

1. Interpolate $\mathbf{u}^n$ onto $\mathbf{p}^{n+1}$ by whatever means to get $\hat{\mathbf{u}}$,

2. Integrate $\hat{\mathbf{u}}$ one time step by computing $\tilde{\mathbf{u}} = \hat{\mathbf{u}} + \Delta t f_{\mathbf{p}^{n+1}}\left(\hat{\mathbf{u}}, \tilde{\mathbf{u}}\right)$,

3. Compute $\mathbf{u}^{n+1}$ by solving the system of $M+1$ equations $\mathbf{u}^{n+1}=\tilde{\mathbf{u}}+\lambda\mathbf{z}$ and $\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1})=\mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)$, for $\mathbf{u}^{n+1}\in\mathbb{R}^M$ and $\lambda\in\mathbb{R}$, where the direction of projection $\mathbf{z}$ is typically an approximation to $\nabla_{\mathbf{u}}\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1})$.

By utilizing the fact that for a method defined by (1.21) there exists an implicitly defined map $\Psi_{\mathbf{p}}:\mathbb{R}^M\to\mathbb{R}^M$ such that $\mathbf{u}^{n+1}=\Psi_{\mathbf{p}}\mathbf{u}^n$, we define

$$g_{\mathbf{p}}(\mathbf{u}^n):=\frac{\Psi_{\mathbf{p}}\mathbf{u}^n-\mathbf{u}^n}{\Delta t},$$

and may then write the tree points above in an equivalent, more compact form as: Compute $\mathbf{u}^{n+1}\in\mathbb{R}^M$ and $\lambda\in\mathbb{R}$ such that

$$\begin{aligned}\mathbf{u}^{n+1}-\hat{\mathbf{u}}-\Delta t g_{\mathbf{p}^{n+1}}\left(\hat{\mathbf{u}}\right)-\lambda\mathbf{z} &= 0, &(1.22)\\ \mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1})-\mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) &= 0, &(1.23)\end{aligned}$$

where $\hat{\mathbf{u}}$ is $\mathbf{u}^n$ interpolated onto $\mathbf{p}^{n+1}$ by an arbitrary procedure.

The following theorem and proof are reminiscent of Theorem 2 and its proof in [23], whose subsequent corollary shows how linear projection methods for solving ODEs are a subset of discrete gradient methods.

Theorem 1.1. *Let $g_{\mathbf{p}}:\mathbb{R}^M\to\mathbb{R}^M$ be a consistent discrete approximation of f in (1.1) and let $\overline{\nabla}\mathcal{I}_{\mathbf{p}}(\mathbf{u}^n,\mathbf{u}^{n+1})$ be any discrete gradient of the consistent approximation $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$ of $\mathcal{I}[u]$ defined by (1.2) on the grid given by discretization parameters $\mathbf{p}$. If we set $S_{\mathbf{p}^{n+1}}$ in (1.20) to be*

$$S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}},\mathbf{u}^{n+1})=\frac{g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\mathbf{z}^T-\mathbf{z}g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})^T}{\left\langle\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}\left(\hat{\mathbf{u}},\mathbf{u}^{n+1}\right),\mathbf{z}\right\rangle},\tag{1.24}$$

then the linear projection method for solving PDEs on a moving grid, given by (1.22)-(1.23), is equivalent to the discrete gradient method on moving grids, as given by (1.20).

Proof. For better readability, take $\overline{\nabla}\mathcal{I}:=\overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}\left(\hat{\mathbf{u}},\mathbf{u}^{n+1}\right)$. Assume that (1.22)-(1.23) are satisfied. By applying (1.23), we get that

$$\begin{aligned}\mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)-\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) &= \mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1})-\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\\ &= \left\langle\overline{\nabla}\mathcal{I},\mathbf{u}^{n+1}-\hat{\mathbf{u}}\right\rangle\\ &= \Delta t\left\langle\overline{\nabla}\mathcal{I},g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\right\rangle+\lambda\left\langle\overline{\nabla}\mathcal{I},\mathbf{z}\right\rangle,\end{aligned}$$

and hence

$$\lambda=\frac{\mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)-\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})}{\left\langle\overline{\nabla}\mathcal{I},\mathbf{z}\right\rangle}-\Delta t\frac{\left\langle\overline{\nabla}\mathcal{I},g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\right\rangle}{\left\langle\overline{\nabla}\mathcal{I},\mathbf{z}\right\rangle}\tag{1.25}$$

Substituting this into (1.22), we get

$$\mathbf{u}^{n+1} = \hat{\mathbf{u}} + \frac{\mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) - \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})}{\left\langle \overline{\nabla}\mathcal{I}, \mathbf{z} \right\rangle}\mathbf{z} + \Delta t \left(g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \frac{\left\langle \overline{\nabla}\mathcal{I}, g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) \right\rangle}{\left\langle \overline{\nabla}\mathcal{I}, \mathbf{z} \right\rangle}\mathbf{z} \right),$$

where

$$\begin{aligned} g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \frac{\left\langle \overline{\nabla}\mathcal{I}, g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) \right\rangle}{\left\langle \overline{\nabla}\mathcal{I}, \mathbf{z} \right\rangle}\mathbf{z} &= \frac{\overline{\nabla}\mathcal{I}^{\mathrm{T}}\mathbf{z} g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \overline{\nabla}\mathcal{I}^{\mathrm{T}} g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\mathbf{z}}{\left\langle \overline{\nabla}\mathcal{I}, \mathbf{z} \right\rangle} \\ &= \frac{g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})\mathbf{z}^{\mathrm{T}}\overline{\nabla}\mathcal{I} - \mathbf{z} g_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}})^{\mathrm{T}}\overline{\nabla}\mathcal{I}}{\left\langle \overline{\nabla}\mathcal{I}, \mathbf{z} \right\rangle} \end{aligned}$$

and thus (1.20) is satisfied, with $S_{\mathbf{p}^{n+1}}$ as given by (1.24). Conversely, if $\mathbf{u}^{n+1}$ satisfies (1.20), then (1.23) is satisfied. Furthermore, inserting (1.24) into (1.20) and following the above deduction backwards, we get (1.22), with λ defined by (1.25). □

Since (1.24) defines a particular set of choices for $S_{\mathbf{p}^{n+1}}$, the linear projection methods on moving grids constitute a subset of all possible discrete gradient methods on moving grids as defined by (1.20). Note also that, since the linear projection methods are independent of the discrete gradient, each linear projection method defines an equivalence class of the methods (1.20), uniquely defined by the choice of $g_{\mathbf{p}^{n+1}}$.

1.3.3 Family of discretized integrals

At the core of the methods considered here is the notion that an approximation to the first integral $\mathcal{I}$ is preserved, and that this approximation is dependent on the discretization parameters which may change from iteration to iteration. That is, we have a family of discretized first integrals $\mathcal{I}_{\mathbf{p}}$, and at each time step the discretized first integral is exchanged for another. For each set of discretization parameters $\mathbf{p}$, there is a corresponding set of degrees of freedom $\mathbf{u}$, in which we search for a $\mathbf{u}$ such that $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$ is preserved. This can be interpreted as a fiber bundle with base space B as the set of all possible discretization parameters $\mathbf{p}$, and fibers $F_{\mathbf{p}}$ as the sets of all degrees of freedom such that the discretized first integral is equal to the initial discretized first integral, i.e. $F_{\mathbf{p}} = \{\mathbf{u} \in \mathbb{R}^M | \mathcal{I}_{\mathbf{p}}(\mathbf{u}) = \mathcal{I}_{\mathbf{p}^0}(\mathbf{u}^0)\}$. A similar idea, although without energy preservation, has been discussed by Bauer, Joshi and Modin in [2].

1.4 Numerical experiments

To provide examples of the application of our method and to investigate its accuracy, we have applied it to two one-dimensional PDEs: the sine-Gordon equation and the Korteweg–de Vries (KdV) equation. The choice of these equations were made because they both possess traveling wave solutions in the form of solitons, providing an ideal situation for r-adaptivity, which allows the grid points to cluster around wave fronts. The following experiments consider r-adaptivity only, and not p- or h-adaptivity. The sine-Gordon equation is solved using the FDM formulation of section 1.2.2, while the KdV equation is solved using the PUM formulation of section 1.2.3.

We wish to compare our methods to standard methods on fixed and adaptive meshes. This gives us four methods to consider: Fixed mesh methods with energy preservation by discrete gradients (DG), adaptive mesh methods with preservation by discrete gradients (DGMM), a non-preserving fixed grid method (MP), and the same method with adaptive mesh (MPMM). The former two methods are those described earlier in the paper, while the latter two are made differently for the two equations. In the sine-Gordon case, we use a finite difference scheme where spatial discretization is done using central finite differences and time discretization using the implicit midpoint rule. In the KdV case, the spatial discretization is performed the same way as for the discrete gradient schemes, while the time discretization is done using the implicit midpoint rule. The mesh adaptivity procedure for the DGMM and MPMM schemes is presented in the next subsection.

The MPMM scheme for the sine-Gordon equation appeared unstable unless restrictively short time steps were used, and the results of those tests are therefore omitted from the following discussion. It is difficult to analyze the MPMM scheme and pinpoint an exact cause for this instability. However, it is worth noting that the other three schemes have preservation properties that should contribute to their stability; the DG and DGMM schemes have energy preservation properties, and the semidiscretization used for the sine-Gordon equation gives rise to a Hamiltonian system of equations which means that the MP scheme, which is symplectic, should perform well. On the other hand, the moving mesh strategy used breaks the symplecticity property in the MPMM scheme; specifically, the transfer strategies as presented in the next subsection do not preserve symplecticity. The results using MPMM for the KdV equation were better, and are presented.

1.4.1 Adaptivity

Concerning adaptivity of the mesh, we used a simple method for r-adaptivity which can be applied to both FDM and FEM problems in one spatial dimension.

When applying moving mesh methods, one can either couple the evolution of the mesh with the PDE to be solved through a Moving Mesh PDE [15] or use the rezoning approach, where function values and grid points are calculated in an intermittent fashion. Since our method is based on having a new set of grid points at each time step, and not coupling the evolution of the mesh to the PDE, the latter approach was used. It is based on an equidistribution principle, meaning that when $\Omega = [a, b]$ is split into M intervals, one requires that

$$\int_{x_i}^{x_{i+1}} \omega(x)\mathrm{d}x = \frac{1}{M}\int_{a}^{b} \omega(x)\mathrm{d}x,$$

where the monitor function ω is a function measuring how densely grid points should lie, based on the value of u. The choice of monitor function is problem dependent, and choosing it optimally may require considerable research. A variety of monitor functions have been studied for certain classes of problems, see e.g. [3, 5]. Through numerical experiments, we found little difference in performance when choosing between monitor functions based on arc-length and curvature, and have in the following used the former, that is, the generalized arc-length monitor function [5]

$$\omega(x) = \sqrt{1 + k^2\left(\frac{\partial u}{\partial x}(x)\right)^2}.$$

Here, the equidistribution principle amounts to requiring that the weighted arc length (in the case $k = 1$ one recovers the usual arc length) of u over each interval is equal. In applications, we only have an approximation of u, meaning ω must be approximated as well; in our case, we have applied a finite difference approximation and obtained approximately equidistributing grids using de Boor's method as explained in [16, pp. 36-38]. We tried different smoothing techniques, including a direct smoothing of the monitor function and an iterative procedure for the regridding by De Boor's method (see e.g. [4, 16, 24]). In the case of the KdV equation, there was little to no improvement using smoothing, but the sine-Gordon experiments showed significant improvement with direct smoothing; i.e., in De Boor's algorithm, we use the smoothed discretized monitor function

$$\bar{\omega}_i = \frac{\omega_{i-1} + 2\omega_i + \omega_{i+1}}{4}.$$

Having obtained the discretization parameters for the current time step, the numerical solution $\mathbf{u}$ from the previous time step must be transferred onto the new set of mesh points. We tested three different ways of doing this, two of which are using linear interpolation and cubic interpolation. The linear interpolation consists of constructing a function $\hat{u}(x)$ which is piecewise linear on each interval $[x_i^n, x_{i+1}^n]$ such that $\hat{u}(x_i^n) = u_i^n$, then evaluating this function at the new mesh points, giving the interpolated values $\hat{u}_i = \hat{u}(x_i^{n+1})$. The cubic interpolation consists of a similar construction, using cubic Hermite splines through the MATLAB function `pchip`. Of these two transfer methods, the cubic interpolation yielded superior results in all cases, and so only results using cubic interpolation are presented. The third way, using preserving transfer as presented in section 1.3.1, applies to the KdV example, where the PUM is used. Here, we found little difference between cubic interpolation and exact transfer, so results are presented using cubic interpolation for the transfer operation here as well.

1.4.2 Sine-Gordon equation

The sine-Gordon equation is a nonlinear hyperbolic PDE in one spatial and one temporal dimension exhibiting soliton solutions, with applications in predicting dislocations in crystals and propagation of fluxons in junctions between superconductors. It is stated in initial value problem form as:

$$\begin{aligned} &u_{tt} - u_{xx} + \sin(u) = 0, \quad (x,t) \in \mathbb{R} \times [0,T], \\ &u(x,0) = f(x), \quad u_t(x,0) = g(x). \end{aligned} \tag{1.26}$$

We consider a finite domain $[-L, L] \times [0, T]$ with periodic boundary conditions $u(-L) = u(L)$ and $u_t(-L) = u_t(L)$. The equation has the first integral

$$\mathcal{I}[u] = \int_{\mathbb{R}} \frac{1}{2}u_t^2 + \frac{1}{2}u_x^2 + 1 - \cos(u)\mathrm{d}x.$$

Introducing $v = u_t$, (1.26) can be rewritten as a first-order system of PDEs:

$$\begin{bmatrix} u_t \\ v_t \end{bmatrix} = \begin{bmatrix} v \\ u_{xx} - \sin(u) \end{bmatrix},$$

with first integral

$$\mathcal{I}[u,v] = \int_{\mathbb{R}} \frac{1}{2}v^2 + \frac{1}{2}u_x^2 + 1 - \cos(u)\mathrm{d}x. \tag{1.27}$$

Finding the variational derivative of this, one can interpret the equation in the form (1.3) with S and $\frac{\delta\mathcal{I}}{\delta u}$ as follows:

$$S = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad \frac{\delta\mathcal{I}}{\delta u}[u,v] = \begin{bmatrix} \sin(u) - u_{xx} \\ v \end{bmatrix}.$$

We will apply the FDM approach presented in section 1.2.2, approximating (1.27) by some quadrature with points $\{x_i\}_{i=0}^M$ and weights $\{\kappa_i\}_{i=0}^M$,

$$\mathcal{I}[u,v] \simeq \sum_{i=0}^{M} \kappa_i \left(\frac{1}{2} v_i^2 + \frac{1}{2} u_{x,i}^2 + 1 - \cos(u_i) \right).$$

In addition, we approximate the spatial derivatives with central differences. At the endpoints, a periodic extension is assumed, yielding the approximation

$$\mathcal{I}_{\mathbf{p}}(\mathbf{u}) = \sum_{i=0}^{M} \kappa_i \left(\frac{1}{2} v_i^2 + \frac{1}{2} \left(\frac{\delta u_i}{\delta x_i} \right)^2 + 1 - \cos(u_i) \right).$$

Here, $\delta w_i = w_{i+1} - w_{i-1}$ denotes central difference, with special cases $\delta u_0 = \delta u_M = u_1 - u_{M-1}$, and $\delta x_0 = \delta x_M = x_1 - x_0 + x_M - x_{M-1}$. Taking the gradient of $\mathcal{I}_{\mathbf{p}}(\mathbf{u})$ and applying the AVF discrete gradient gives

$$\overline{\nabla} \mathcal{I}_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}) = \int_0^1 \nabla_{\mathbf{u}} \mathcal{I}_{\mathbf{p}}(\xi \mathbf{u}^n + (1-\xi)\mathbf{u}^{n+1}) \mathrm{d}\xi$$

The periodic boundary conditions are enforced by setting $u_0 = u_M$. In the implementation, the κ_i were chosen as the quadrature weights associated with the composite trapezoidal rule, i.e.

$$\kappa_0 = \frac{x_1 - x_0}{2}, \quad \kappa_M = \frac{x_M - x_{M-1}}{2}, \quad \kappa_i = \frac{x_{i+1} - x_{i-1}}{2}, \quad i = 1, ..., M-1.$$

Furthermore, S was approximated by the matrix

$$S_d = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix},$$

with I an $M \times M$ identity matrix. The exact solution considered was

$$u(x,t) = 4 \tan^{-1} \left(\frac{\sinh\left(\frac{ct}{\sqrt{1-c^2}} \right)}{c \cosh\left(\frac{x}{\sqrt{1-c^2}} \right)} \right).$$

This is a *kink-antikink* system, an interaction between two solitons, each moving in different directions with speed $c \in (0,1)$, resulting in two wave fronts traveling in opposite directions. The wave fronts become steeper as $c \to 1$. Figure 1.1 illustrates the analytical solution and shows the time evolution of the

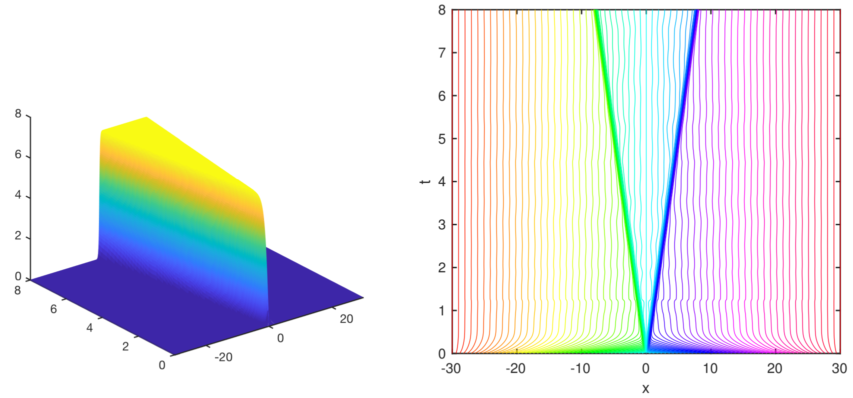


Figure 1.1: *Left*: Illustration of kink-antikink solution. *Right*: Grid movement - each line represents the path of one grid point in time.

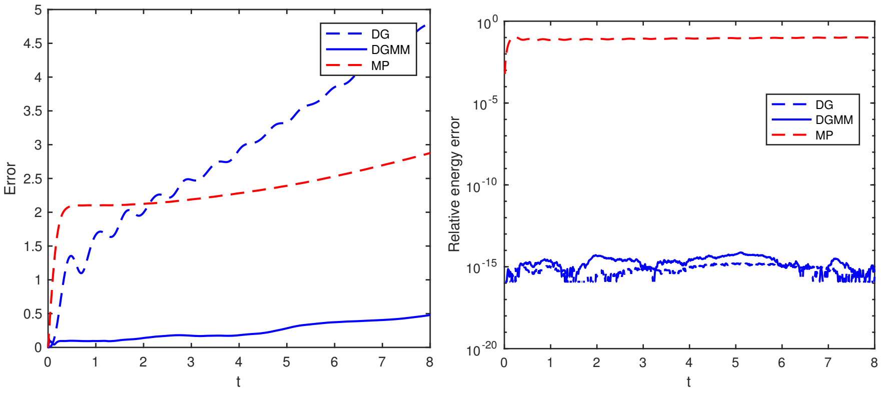


Figure 1.2: *Left*: L_2 error. *Right*: Relative error in $I_{\mathbf{p}}$. *Parameters*: $\Delta t = 0.01$, $M = 300$, $L = 30$, $c = 0.99$.

mesh as obtained with the DGMM method. Note that the grid points cluster along the wave fronts.

The left hand side of Figure 1.2 shows the time evolution of the error $E_n^u = ||u_n^I(x) - u(x, t_n)||_{L_2}$, where u_n^I is a linear interpolant created from the pairs $(\mathbf{u}^n, \mathbf{x}^n)$. The right hand side of Figure 1.2 shows the time evolution of the relative error in the discretized energy, $E_n^I = (I_{\mathbf{p}^n}(\mathbf{u}^n) - I_{\mathbf{p}^0}(\mathbf{u}^0)) / I_{\mathbf{p}^0}(\mathbf{u}^0)$. We can see that the long-term behaviour of the MP scheme is superior to that of the DG scheme, but when mesh adaptivity is applied, the DGMM scheme is clearly better. Also note that while the DG and DGMM schemes preserve $I_{\mathbf{p}}$ to machine precision, the MP scheme does not.

Figure 1.3 shows the convergence behaviour of the three schemes with

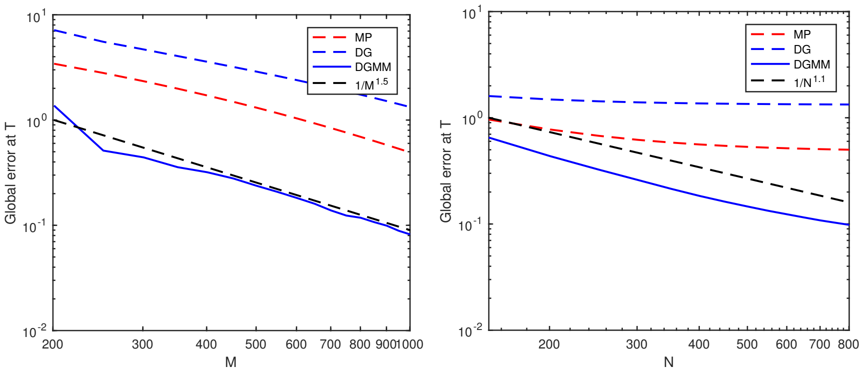


Figure 1.3: *Left*: Error at $T = 8$ as a function of M, with $\Delta t = 0.008$, $c = 0.99$, $L = 30$. *Right*: Error at $T = 8$ as a function of $N = T/\Delta t$, with $M = 1000$, $c = 0.99$, $L = 30$.

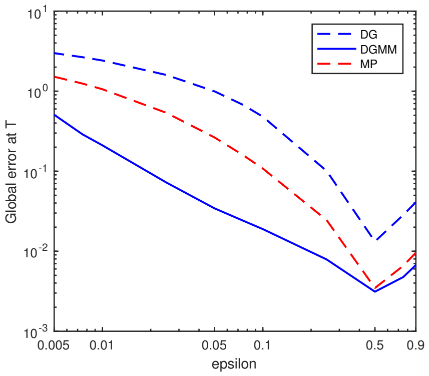


Figure 1.4: Error at $T = 8$ as a function of ϵ, with $\Delta t = 0.01$, $M = 600$ and $L = 30$.

respect to the number of spatial discretization points M, and the number of time steps N. Note that the DG and MP methods plateau at $N \simeq 400$; this is due to the error stemming from spatial discretization dominating the time discretization error for these methods, while the DGMM scheme has lower spatial discretization error. The convergence order of the DGMM scheme was measured using a first order polynomial fitting of $\log(E_n^u)$ to $\log(M)$ and $\log(N)$. The convergence order with respect to M was calculated as 1.518, and the convergence order with respect to N was measured at 1.121.

Finally, to illustrate the applicability of the DGMM scheme to harder problems, Figure 1.4 shows the error at stopping time of the methods as a function of a parameter ϵ representing the increasing speed of the solitons ($c = 1 - \epsilon$). From this plot, it is appararent that while the non-adaptive MP scheme is competitive at low speeds, the moving mesh method provides significantly more accuracy as $c \to 1$.

1.4.3 Korteweg–de Vries equation

The KdV equation is a nonlinear PDE with soliton solutions modelling shallow water surfaces, stated as

$$u_t + u_{xxx} + 6uu_x = 0. \tag{1.28}$$

It has infinitely many first integrals, one of which is the Hamiltonian

$$\mathcal{H}[u] = \int\limits_{\mathbb{R}} \frac{1}{2}u_x^2 - u^3 \mathrm{d}x.$$

With this Hamiltonian, we can write (1.28) in the form (1.3) with S and $\frac{\delta\mathcal{H}}{\delta u}$ as follows:

$$S = \frac{\partial}{\partial x}, \quad \frac{\delta\mathcal{H}}{\delta u}[u] = -u_{xx} - 3u^2.$$

We will apply the PUM approach to create a numerical scheme which preserves an approximation to $\mathcal{H}[u]$, splitting $\Omega = [-L, L]$ into M elements $\{[x_i, x_{i+1}]\}_{i=0}^{M-1}$ and using Lagrangian basis functions φ_j of arbitrary degree for the trial space. Approximating u by u^h as in section 1.2.3, we find

$$\begin{aligned}\mathcal{H}_{\mathbf{p}}(\mathbf{u}) = \mathcal{H}[u^h] &= \int_\Omega \frac{1}{2}(u_x^h)^2 - (u^h)^3 \mathrm{d}x \\ &= \frac{1}{2}\sum_{j,k} u_j u_k \int_\Omega \varphi_{j,x}\varphi_{k,x}\mathrm{d}x - \sum_{j,k,l} u_j u_k u_l \int_\Omega \varphi_j\varphi_k\varphi_l \mathrm{d}x. \end{aligned} \tag{1.29}$$

The integrals can be evaluated exactly and efficiently by considering elementwise which basis functions are supported on the element before applying Gaussian quadrature to obtain exact evaluations of the polynomial integrals. We define

$$D_{ijk} = \int_\Omega \varphi_i\varphi_j\varphi_k \mathrm{d}x \quad \text{and} \quad E_{ij} = \int_\Omega \varphi_{i,x}\varphi_{j,x}\mathrm{d}x.$$

The matrices A and B with

$$A_{ij} = \int_\Omega \varphi_i\varphi_j \mathrm{d}x \quad \text{and} \quad B_{ji} = \int_\Omega \varphi_i\varphi_{j,x}\mathrm{d}x$$

are formed in the same manner. Note that B is in this case independent of $\mathbf{u}$. Applying the AVF method yields the discrete gradient

$$\overline{\nabla}\mathcal{H}_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}) = \int\limits_0^1 \nabla_{\mathbf{u}}\mathcal{H}_{\mathbf{p}}(\xi\mathbf{u}^n + (1-\xi)\mathbf{u}^{n+1})\mathrm{d}\xi$$

such that, with the convention of summation over repeated indices,

$$(\overline{\nabla}\mathcal{H}_{\mathbf{p}})_i = \frac{1}{2}E_{ij}(u_j^n + u_j^{n+1}) - D_{ijk}(u_j^n(u_k^n + \frac{1}{2}u_k^{n+1}) + u_j^{n+1}(\frac{1}{2}u_k^n + u_k^{n+1})).$$

This gives us all the required terms for forming the system (1.15) and applying the discrete gradient method to it. During testing, the φ_j were chosen as piecewise linear polynomials. The exact solution considered is of the form

$$u(x,t) = \frac{c}{2}\text{sech}^2\left(\frac{\sqrt{c}}{2}(x-ct)\right), \tag{1.30}$$

which is a right-moving soliton with c as the propagation speed, chosen as $c = 6$ in the numerical tests. We have considered periodic boundary conditions on a domain $[-L, L] \times [0, T]$, with $L = 100$ in all the following results.

Our discrete gradient method on a moving mesh (DGMM) is compared to the same method on a static, equidistributed mesh (DG), and the implicit midpoint method on static (MP) and moving mesh (MPMM). The spatial discretization is performed the same way in all cases. Figure 1.5 shows an example of exact and numerical solutions at $t = 15$. Note that the peak in the exact solution will be located at $x = ct$.

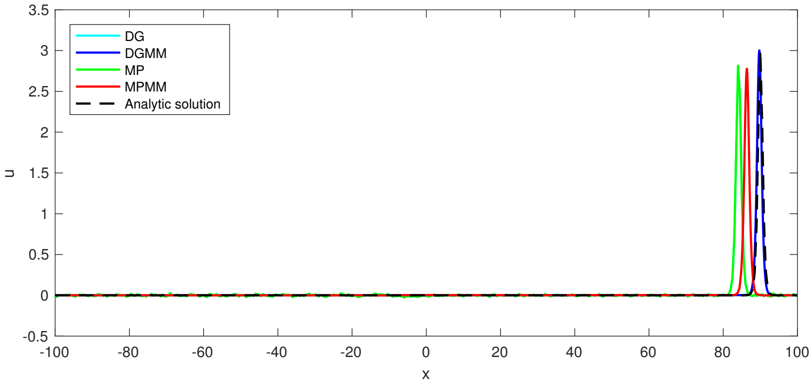


Figure 1.5: Solutions at T = 15. $\Delta t = 0.01$, $M = 400$. MP and DG are almost indistinguishable.

To evaluate the numerical solution, it is reasonable to look at the distance error

$$E_n^{\text{dist}} = ct_n - x^*,$$

where $x^* = \arg\max_x u_h(x, t_n)$, i.e. the location of the peak in the numerical solution. Another measure of the error is the shape error

$$E_n^{\text{shape}} = \left\| u_h(x,t_n) - u\left(x, \frac{x^*}{c}\right) \right\|,$$

where the peak of the exact solution is translated to match the peak of the numerical solution, and the shapes of the solitons are compared.

Figure 1.6 confirms that the DG and DGMM methods preserve the approximated Hamiltonian (1.29), while it is also worth noting that in the case of the midpoint method, the error in this conserved quantity is much larger on a moving than on a static mesh. Similar behaviour is also observed for a moving-mesh method for the regularized long wave equation in the recent paper [19], where it is concluded that a moving mesh method with a conservative property would be an interesting research topic. Figure 1.7, where the phase and shape errors are plotted up to $T = 15$, is an example of how the DGMM method performs comparatively better with increasing time.

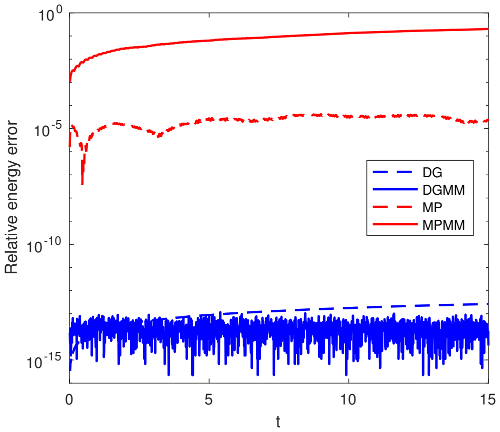


Figure 1.6: Relative error in the Hamiltonian plotted as a function of time $t \in [0, 15]$. $\Delta t = 0.01$, $M = 400$.

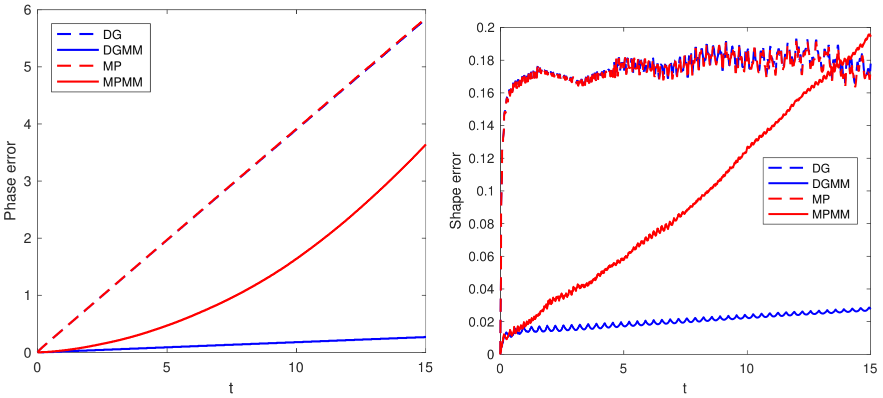


Figure 1.7: Phase error (left) and shape error (right) as a function of time. $\Delta t = 0.01$, $M = 400$.

In figures 1.8 and 1.9 we present the phase and shape errors for the different methods as a function of the number of elements M and the number of time steps N, respectively. Reference lines are included to give an indication of the rate of convergence. We also calculated this for the DGMM method by first degree polynomial fitting of the error curve, giving a convergence order of 1.135 for the phase error and 2.311 for the shape error as a function of M. As a function of N, we get a convergence order of 1.492 for the phase error, and 1.609 for the shape error (the latter measured up to $N = 320$, where it flattens out). We observe that the DGMM scheme performs especially well, compared to the other three schemes, for a coarse spatial discretization compared to the discretization in time.

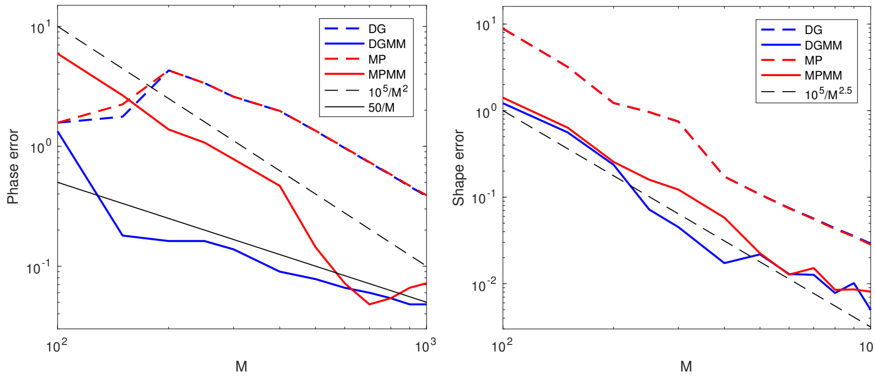


Figure 1.8: Phase error (left) and shape error (right) as a function of the number of elements M, at time $T = 5$. $\Delta t = 0.01$.

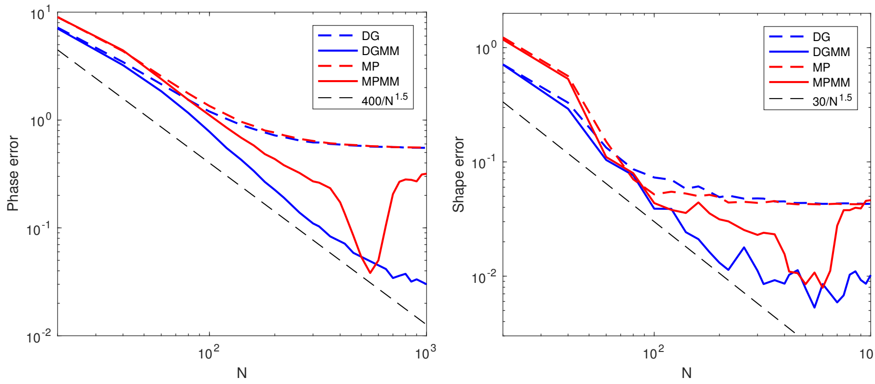


Figure 1.9: Phase error (left) and shape error (right) at time $T = 5$, as a function of the number of time steps $N = T/\Delta t$. $M = 800$.

In figure 1.10, the phase and shape errors are plotted as a function of the parameter c in the exact solution (1.30), where we note that $\frac{c}{2}$ is the height of the wave; increasing c leads to sharper peaks and thus a harder numerical problem. As expected, the advantages of the DGMM method is less evident for small c, but we observe that the DGMM method outperforms the static grid midpoint method already when $c = 2$.

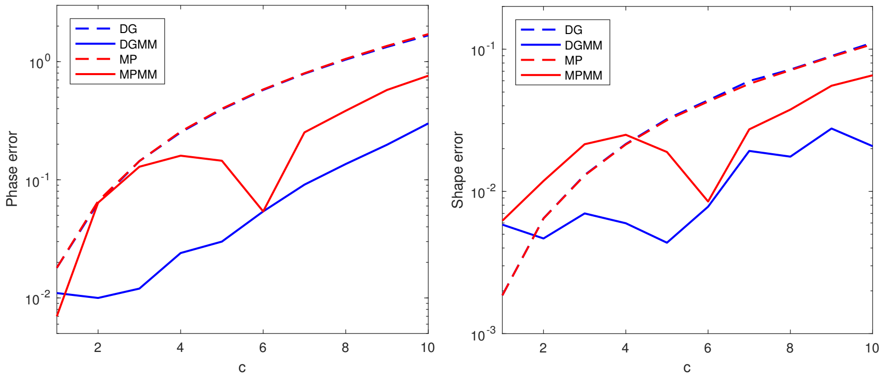


Figure 1.10: Phase error (left) and shape error (right) as a function of c in the exact solution (1.30), at time $t = 5$. $\Delta t = 0.01$, $M = 800$.

1.4.4 Execution time

The code used is not optimized, so any quantitative comparison to standard methods has not been performed; it is still possible to make some qualitative observations. Adding adaptivity increases time per iteration slightly since the systems become more complicated, especially in the case of the PUM approach where the matrices A and B need to be recalculated, at each time step when adaptivity is used. This increases runtime somewhat when compared to fixed grid methods. However, adaptivity allows for using fewer degrees of freedom, and so decreases the degrees of freedom needed for a given level of accuracy. This accuracy gain is more pronounced the harder the problem is (steeper wave fronts etc.), and so it stands to reason that there will be situations where adaptive energy preserving methods will outperform non-adaptive and/or non-preserving methods.

1.5 Conclusion

In this paper, we have introduced a general framework for producing adaptive first integral preserving methods for partial differential equations. This is done by first providing two means of producing first integral preserving methods on arbitrary fixed grids, then showing how to extend these methods to allow for adaptivity while preserving the first integral. Numerical testing shows that moving mesh methods coupled with discrete gradient methods provide good solvers for the sine-Gordon and Korteweg–de Vries equations. It would be of interest to apply the method to higher-dimensional PDEs with a more challenging geometry, preferably using the PUM approach, to investigate its accuracy as compared to conventional methods, and to test whether h- and/or p-refinement provides a notable improvement. It may also prove fruitful to explore the ideas presented in [2] to make the transfer operations between sets of discretization parameters in a more natural setting than simply interpolating, as suggested in section 1.3.3. Furthermore, analysis of the methods considered here could provide important insight into e.g. stability, consistency and convergence order.

Bibliography

[1] I. Babuška and B. Guo. The h, p and h-p version of the finite element method; basis theory and applications. *Adv. Eng. Softw.*, 15:159–174, 1992.

[2] M. Bauer, S. Joshi, and K. Modin. Diffeomorphic density matching by optimal information transport. *SIAM J. Imaging Sci.*, 8(3):1718–1751, 2015.

[3] J. Blom and J. Verwer. On the use of the arclength and curvature monitor in a moving-grid method which is based on the method of lines. Technical report, NM-N8902, CWI, Amsterdam, 1989.

[4] C. J. Budd, W. Huang, and R. D. Russell. Moving mesh methods for problems with blow-up. *SIAM J. Sci. Comput.*, 17(2):305–327, 1996.

[5] C. J. Budd, W. Huang, and R. D. Russell. Adaptivity with moving grids. *Acta Numer.*, 18:111–241, 2009.

[6] 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(20):6770–6789, 2012.

[7] S. H. Christiansen, H. Z. Munthe-Kaas, and B. Owren. Topics in structure-preserving discretization. *Acta Numer.*, 20:1–119, 2011.

[8] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Ann.*, 100(1):32–74, 1928.

[9] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33(5):2318–2340, 2011.

[10] D. Furihata. Finite difference schemes for $\partial u/\partial t = (\partial/\partial x)^{\alpha}\delta G/\delta u$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156(1):181–205, 1999.

[11] D. Furihata. Finite-difference schemes for nonlinear wave equation that inherit energy conservation property. *J. Comput. Appl. Math.*, 134(1-2):37–57, 2001.

[12] D. Furihata and T. Matsuo. *Discrete variational derivative method*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011.

[13] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[14] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer, Heidelberg, 2010.

[15] W. Huang and R. Russell. Adaptive mesh movement - the MMPDE approach and its applications. *J. Comput. Appl. Math.*, 128:383–398, 2001.

[16] W. Huang and R. Russell. *Adaptive moving mesh methods*, volume 174 of *Springer Series in Applied Mathematical Sciences*. Springer-Verlag, New York, 2010.

[17] T. Lee, M. Baines, and S. Langdon. A finite difference moving mesh method based on conservation for moving boundary problems. *J. Comput. Appl. Math.*, 288:1–17, 2015.

[18] S. Li and L. Vu-Quoc. Finite difference calculus invariant structure of a class of algorithms for the nonlinear Klein-Gordon equation. *SIAM J. Numer. Anal.*, 32(6):1839–1875, 1995.

[19] C. Lu, W. Huang, and J. Qiu. An adaptive moving mesh finite element solution of the regularized long wave equation. *J. Sci. Comput.*, 74(1):122–144, 2018.

[20] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *Philos. T. R. Soc. A*, 357(1754):1021–1045, 1999.

[21] J. Melenk and I. Babuška. The partition of unity finite element method: Basic theory and applications. *Comput. Method. Appl. M.*, 139:289–314, 1996.

[22] Y. Miyatake and T. Matsuo. A note on the adaptive conservative/dissipative discretization for evolutionary partial differential equations. *J. Comput. Appl. Math.*, 274:79–87, 2015.

[23] R. A. Norton, D. I. McLaren, G. R. W. Quispel, A. Stern, and A. Zanna. Projection methods and discrete gradient methods for preserving first integrals of ODEs. *Discrete Contin. Dyn. Syst.*, 35(5):2079–2098, 2015.

[24] J. D. Pryce. On the convergence of iterated remeshing. *IMA J. Numer. Anal.*, 9(3):315–335, 1989.

[25] G. R. W. Quispel and D. I. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A: Math. Theor.*, 41(045206), 2008.

[26] R. D. Richtmyer and K. W. Morton. *Difference methods for initial-value problems*. Second edition. Interscience Tracts in Pure and Applied Mathematics, No. 4. Interscience Publishers John Wiley & Sons, Inc., New York-London-Sydney, 1967.

[27] T. Yaguchi, T. Matsuo, and M. Sugihara. An extension of the discrete variational method to nonuniform grids. *J. Comput. Phys.*, 229(11):4382–4423, 2010.

[28] T. Yaguchi, T. Matsuo, and M. Sugihara. The discrete variational derivative method based on discrete differential forms. *J. Comput. Phys.*, 231(10):3963–3986, 2012.

[29] P. A. Zegeling. *r*-refinement for evolutionary PDEs with finite elements or finite differences. *Appl. Numer. Math.*, 26:97–104, 1998.

Energy preserving moving mesh methods applied to the BBM equation

Sølve Eidnes and Torbjørn Ringholm

Published in Proceedings of MekIT '17, pp. 121–136 (2017)

Energy preserving moving mesh methods applied to the BBM equation

Abstract. Energy preserving numerical methods for a certain class of PDEs are derived, applying the partition of unity method. The methods are extended to also be applicable in combination with moving mesh methods by the rezoning approach. These energy preserving moving mesh methods are then applied to the Benjamin–Bona–Mahony equation, resulting in schemes that exactly preserve an approximation to one of the Hamiltonians of the system. Numerical experiments that demonstrate the advantages of the methods are presented.

2.1 Introduction

Numerical solutions of differential equations by standard methods will typically not inherit invariant properties from the original, continuous problem. Since the energy-preserving methods of Courant, Friedrichs and Lewy were introduced in [8], the development of conservative methods has garnered much interest and considerable research, surveyed in [15] up to the early 1990s. In some important cases, conservation properties can be used to ensure numerical stability or existence and uniqueness of the numerical solution. In other cases, the conservation of one or more invariants can be of importance in its own right. In addition, as noted in [13], one may expect that when properties of the continuous dynamical system are inherited by the discrete dynamical system, the numerical solution can be more accurate, especially over large time intervals.

The discrete gradient methods for ordinary differential equations (ODEs), usually attributed to Gonzalez [12], are methods that preserve first integrals exactly. Since the late 1990s, a number of researchers have worked on extending this theory to create a counterpart for partial differential equations (PDEs), see e.g. [5, 11]. Such methods, which are either called discrete variational derivative methods or discrete gradient methods for PDEs, aim at preserving some discrete approximation of a first integral which is preserved by the continuous system. Up to very recently, the schemes presented have typically been based on a finite difference approach, and exclusively on fixed, uniform grids. Two different discrete variational derivative methods on fixed, non-uniform grids were presented by Yaguchi, Matsuo and Sugihara in [21, 22]. In [18], Miyatake and Matsuo introduce integral preserving methods on adaptive grids for certain classes of PDEs. Eidnes, Owren and Ringholm presented in [10] a general approach to extending the theory of discrete variational derivative methods, or discrete gradient methods for PDEs, to adaptive grids, using either a finite difference approach, or the partition of unity method, which can be seen as a generalization of the finite element method.

In this paper, we present an application of the approach introduced in [10] to the Benjamin–Bona–Mahony (BBM) equation, also called the regularized long wave equation in the literature. Although what we present here is a finite element method, the theory can be easily applied in a finite difference setting. Previously, there have been developed integral preserving methods for this equation [6], as well as adaptive moving mesh methods [16], but the schemes we are to present here are, to our knowledge, the first combining these properties. In fact, in [16] it is noted that combining integral preservation with adaptivity is an interesting topic for further research.

2.2 The discrete gradient methods for PDEs

We give a quick survey of the discrete gradient methods for PDEs, and present an approach to the spatial discretization by the partition of unity method (PUM).

2.2.1 Problem statement

Consider a PDE of the form

$$u_t = f(\mathbf{x}, u^J), \qquad \mathbf{x} \in \Omega \subseteq \mathbb{R}^d, \quad u \in \mathcal{B} \subseteq L^2, \tag{2.1}$$

where u^J denotes u itself and its partial derivatives of any order with respect to the spatial variables $x_1, ..., x_d$, and where we assume that $\mathcal{B}$ is sufficiently regular to allow all operations used in the following.

We define a *first integral* of (2.1) to be a functional $\mathcal{I}[u]$ satisfying

$$\left\langle \frac{\delta \mathcal{I}}{\delta u}[u], f(\mathbf{x}, u^J) \right\rangle_{L^2} = 0, \quad \forall u \in \mathcal{B},$$

recalling that the *variational derivative* $\frac{\delta \mathcal{I}}{\delta u}[u]$ is defined as the function satisfying

$$\left\langle \frac{\delta \mathcal{I}}{\delta u}[u], v \right\rangle_{L^2} = \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \mathcal{I}[u + \epsilon v] \quad \forall v \in \mathcal{B}.$$

This means that $\mathcal{I}[u]$ is preserved over time by (2.1), since

$$\frac{\mathrm{d}\mathcal{I}}{\mathrm{d}t} = \left\langle \frac{\delta \mathcal{I}}{\delta u}[u], \frac{\partial u}{\partial t} \right\rangle_{L^2} = 0.$$

Furthermore, we may observe that if there exists some operator $S(\mathbf{x}, u^J)$, skew-symmetric with respect to the L^2 inner product, such that

$$f(\mathbf{x}, u^J) = S(\mathbf{x}, u^J)\frac{\delta\mathcal{I}}{\delta u}[u],$$

then $\mathcal{I}[u]$ is a first integral of (2.1), and we can state (2.1) on the form

$$u_t = S(\mathbf{x}, u^J)\frac{\delta\mathcal{I}}{\delta u}[u]. \tag{2.2}$$

The idea behind the discrete variational derivative methods is to derive a discrete version of the PDE on the form (2.2), by obtaining a so-called discrete variational derivative and approximate $S(\mathbf{x}, u^J)$ by a skew-symmetric matrix, see e.g. [11].

As proven in [10], all discrete variatonal derivative methods can be expressed as discrete gradient methods on a system of ODEs obtained by discretizing (2.2) in space, to get a system

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = S(\mathbf{u})\nabla I(\mathbf{u}), \tag{2.3}$$

where $S(\mathbf{u})$ is a skew-symmetric matrix. The discrete gradient methods for such a system of ODEs preserve the first integral $I(\mathbf{u})$ [17]. These numerical methods are given by

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \bar{S}(\mathbf{u}^n, \mathbf{u}^{n+1})\overline{\nabla}I(\mathbf{u}^n, \mathbf{u}^{n+1}),$$

where $\bar{S}(\mathbf{u}^n, \mathbf{u}^{n+1})$ is a consistent skew-symmetric time-discrete approximation to $S(\mathbf{u})$ and $\overline{\nabla}I(\mathbf{v}, \mathbf{u})$ is a discrete gradient of $I(\mathbf{u})$, defined as a function satisfying

$$\begin{aligned}(\overline{\nabla}I(\mathbf{v}, \mathbf{u}))^T(\mathbf{u} - \mathbf{v}) &= I(\mathbf{u}) - I(\mathbf{v}),\\ \overline{\nabla}I(\mathbf{u}, \mathbf{u}) &= \nabla I(\mathbf{u}).\end{aligned}$$

There are many possible choices of discrete gradients. For the numerical experiments in this note, we will use the Average Vector Field (AVF) discrete gradient [5], given by

$$\overline{\nabla}I(\mathbf{v}, \mathbf{u}) = \int_0^1 \nabla I(\xi\mathbf{u} + (1 - \xi)\mathbf{v})\mathrm{d}\xi,$$

Note that when discretizing the system (2.2) in space, we do so by finding a discrete approximation $\mathcal{I}_\mathbf{p}$ to the integral $\mathcal{I}$, and define an energy preserving method to be a method preserving this approximation.

2.2.2 Partition of unity method on a fixed mesh

The partition of unity method is a generalization of the finite element method (FEM). Stating a weak form of (2.2), the problem consists of finding $u \in \mathcal{B}$ such that

$$\langle u_t, v \rangle_{L^2} = \left\langle S(\mathbf{x}, u^J) \frac{\delta \mathcal{I}}{\delta u}[u], v \right\rangle_{L^2} = -\left\langle \frac{\delta \mathcal{I}}{\delta u}[u], S(\mathbf{x}, u^J) v \right\rangle_{L^2} \qquad \forall v \in \mathcal{B}.$$

We define an approximation to u by

$$u^h(x,t) = \sum_{i=0}^{M} u_i(t) \varphi_i(x),$$

where the test functions $\varphi_i(x)$ span a finite-dimensional subspace $\mathcal{B}^h \subseteq \mathcal{B}$. Referring to [10] for details, we then obtain the Galerkin form of the problem: Find $u_i(t), i = 0, \ldots, M,$ such that

$$\sum_{i=0}^{M} \frac{\mathrm{d}u_i}{\mathrm{d}t} \left\langle \varphi_i, \varphi_j \right\rangle_{L^2} = -\sum_{i=0}^{M} w_i(\mathbf{u}) \left\langle \varphi_i, S(\mathbf{x}, u^{h,J}) \varphi_j \right\rangle_{L^2} \qquad \forall j \in \{0, ..., M\},$$

where, with $A_{ij} = \left\langle \varphi_i, \varphi_j \right\rangle_{L^2}$,

$$\mathbf{w}(\mathbf{u}) = A^{-1} \nabla \mathcal{I}_{\mathbf{p}}(\mathbf{u}).$$

We end up with an ODE for the coefficients u_i:

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = S_{\mathbf{p}}(\mathbf{u}) \nabla \mathcal{I}_{\mathbf{p}}(\mathbf{u}). \tag{2.4}$$

Here, $S_{\mathbf{p}}(\mathbf{u}) = -A^{-1} B(\mathbf{u}) A^{-1}$ is a skew-symmetric matrix, with $B(\mathbf{u})$ given by $B(\mathbf{u})_{ji} = \left\langle \varphi_i, S(\mathbf{x}, u^{h,J}) \varphi_j \right\rangle_{L^2}$, and the system is thereby of the form (2.3). Then, the scheme

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = S_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}) \overline{\nabla} \mathcal{I}_{\mathbf{p}}(\mathbf{u}^n, \mathbf{u}^{n+1}).$$

will preserve the approximated first integral $\mathcal{I}_{\mathbf{p}}$ in the sense that $\mathcal{I}_{\mathbf{p}}(\mathbf{u}^{n+1}) = \mathcal{I}_{\mathbf{p}}(\mathbf{u}^n)$.

2.3 Adaptive schemes

The primary motivation for using an adaptive mesh is usually to increase accuracy while keeping computational cost low, by improving discretization locally.

Such methods are typically useful for problems with e.g. traveling wave solutions and boundary layers. The different strategies for adaptive meshes can be classified into two main groups [14]: The quasi-Lagrange approach involves coupling the evolution of the mesh with the PDE, and then solving the problems simultaneously; The rezoning approach consists of calculating the function values and mesh points in an intermittent fashion. Our method can be coupled with any adaptive mesh strategy utilizing the latter approach.

2.3.1 Adaptive discrete gradient methods

Let $\mathbf{p}^n$, $\mathbf{u}^n$, $\mathbf{p}^{n+1}$, and $\mathbf{u}^{n+1}$ denote the discretization parameters and the numerical values obtained at the current time step and next time step, respectively. Note that we now alter the notion of a preserved first integral further, to requiring that $\mathcal{I}_{\mathbf{p}^{n+1}}(\mathbf{u}^{n+1}) = \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)$. The idea behind our approach is to find $\mathbf{p}^{n+1}$ based on $\mathbf{u}^n$ and $\mathbf{p}^n$, transfer $\mathbf{u}^n$ to $\mathbf{p}^{n+1}$ to obtain $\hat{\mathbf{u}}$, and then use $\hat{\mathbf{u}}$ to propagate in time to get $\mathbf{u}^{n+1}$. If the transfer operation between the meshes is preserving, i.e. if $\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) = \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n)$, then the next time step can be taken with the discrete gradient method for static meshes. If, however, non-preserving transfer is used, corrections are needed in order to get a numerical scheme. We introduce in [10] the scheme

$$\mathbf{u}^{n+1} = \hat{\mathbf{u}} - \frac{(\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n))\mathbf{z}}{\left\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}, \mathbf{u}^{n+1}), \mathbf{z} \right\rangle} + \Delta t S_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}, \mathbf{u}^{n+1}) \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}, \mathbf{u}^{n+1}), \tag{2.5}$$

where $\mathbf{z}$ is a vector which should be chosen so as to obtain a minimal correction, and such that $\langle \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}, \mathbf{u}^{n+1}), \mathbf{z} \rangle \neq 0$. In the numerical experiments to follow, we have used $\mathbf{z} = \overline{\nabla}\mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}, \mathbf{u}^{n+1})$.

A preserving transfer can by obtained using the method of Lagrange multipliers. Depending on whether r- p- or h-refinement (or a combination) is used between time steps, we expect the shape and/or number of basis functions to change. See e.g. [14] or [1] for examples of how the basis may change through adaptivity. Denote by $\mathcal{B}^h = \mathrm{span}\{\varphi_i\}_{i=0}^{M}$ the trial space from the current time step and by $\hat{\mathcal{B}}^h = \mathrm{span}\{\hat{\varphi}_i\}_{i=0}^{\hat{M}}$ the trial space for the next time step, and note that in general, $M \neq \hat{M}$. We wish to transfer the approximation u^h from $\mathcal{B}^h$ to $\hat{\mathcal{B}}^h$, obtaining an approximation $\hat{u}^h$, while conserving the first integral, i.e. $\mathcal{I}[u^h] = \mathcal{I}[\hat{u}^h]$. This can be formulated as a constrained minimization problem:

$$\min_{\hat{u}^h \in \tilde{\mathcal{B}}^\langle} ||\hat{u}^h - u^h||_{L^2}^2 \quad \text{s.t.} \quad \mathcal{I}[\hat{u}^h] = \mathcal{I}[u^h]. \tag{2.6}$$

Observe that

$$\begin{aligned} ||\hat{u}^h - u^h||_{L^2}^2 &= \sum_{i=0}^{\hat{M}}\sum_{j=0}^{\hat{M}} \hat{u}_i \hat{u}_j \hat{A}_{ij} - 2\sum_{i=0}^{\hat{M}}\sum_{j=0}^{M} \hat{u}_i u_j^n C_{ij} + \sum_{i=0}^{M}\sum_{i=0}^{M} u_i^n u_j^n A_{ij} \\ &= \hat{\mathbf{u}}^T \hat{A} \hat{\mathbf{u}} - 2\hat{\mathbf{u}}^T C \mathbf{u}^n + \mathbf{u}^n A \mathbf{u}^n, \end{aligned}$$

where $A_{ij} = \langle \varphi_i, \varphi_j \rangle_{L^2}$, $\hat{A}_{ij} = \langle \hat{\varphi}_i, \hat{\varphi}_j \rangle_{L^2}$ and $C_{ij} = \langle \hat{\varphi}_i, \varphi_j \rangle_{L^2}$. The problem (2.6) can thus be reformulated as

$$\min_{\hat{\mathbf{u}} \in \mathbb{R}^{\hat{M}+1}} \hat{\mathbf{u}}^T \hat{A} \hat{\mathbf{u}} - 2\hat{\mathbf{u}}^T C \mathbf{u}^n + \mathbf{u}^n A \mathbf{u}^n \quad \text{s.t.} \quad \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) = 0.$$

This is a quadratic minimization problem with one nonlinear equality constraint, for which the solution $\hat{\mathbf{u}}$ is the solution of the nonlinear system of equations

$$\begin{aligned} \hat{A}\hat{\mathbf{u}} - C\mathbf{u}^n - \lambda \nabla \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) &= 0 \\ \mathcal{I}_{\mathbf{p}^{n+1}}(\hat{\mathbf{u}}) - \mathcal{I}_{\mathbf{p}^n}(\mathbf{u}^n) &= 0, \end{aligned}$$

which can be solved numerically using a suitable nonlinear solver.

2.4 Adaptive energy preserving schemes for the BBM equation

2.4.1 The BBM equation

The BBM equation was introduced by Peregrine [19], and later studied by Benjamin et al. [2] as a model for small amplitude long waves on the surface of water in a channel. Conservative finite difference schemes for the BBM equation were proposed in [20] and [6], the latter being a discrete gradient method on fixed grids. A moving mesh FEM scheme employing a quasi-Lagrange approach is presented by Lu, Huang and Qiu in [16], which we also refer to for a more extensive list of references to the existing numerical schemes for the BBM equation.

Consider now an initial-boundary value problem of the one-dimensional BBM equation with periodic boundary conditions,

$$u_t - u_{xxt} + u_x + uu_x = 0, \qquad x \in [-L, L], \quad t \in (0, T] \tag{2.7}$$

$$u(x, 0) = u_0(x), \qquad x \in [-L, L] \tag{2.8}$$

$$u(-L, t) = u(L, t), \qquad t \in (0, T]. \tag{2.9}$$

By introducing the new variable $m(x,t) coloneqq u(x,t) - u_{xx}(x,t)$, equation (2.7) can be rewritten on the form (2.2) as

$$m_t = \mathcal{S}(m)\frac{\delta \mathcal{H}}{\delta m},$$

for two different pairs of an antisymmetric differential operator $\mathcal{S}(m)$ and a Hamiltonian $\mathcal{H}[m]$:

$$\begin{aligned}\mathcal{S}^1(m) &= -(\frac{2}{3}u+1)\partial_x - \frac{1}{3}u_x,\\ \mathcal{H}^1[m] &= \frac{1}{2}\int (u^2+u_x^2)\mathrm{d}x,\end{aligned}$$

and

$$\begin{aligned}\mathcal{S}^2(m) &= -\partial_x + \partial_{xxx},\\ \mathcal{H}^2[m] &= \frac{1}{2}\int (u^2+\frac{1}{3}u^3)\mathrm{d}x.\end{aligned}$$

2.4.2 Discrete schemes

We apply the PUM approach to create numerical schemes which preserve an approximation to either $\mathcal{H}^1[m]$ or $\mathcal{H}^2[m]$, splitting $\Omega coloneqq[-L,L]$ into M elements $\{[x_i,x_{i+1}]\}_{i=0}^{M-1}$. Defining the matrices A and E by their components

$$A_{ij} = \int_\Omega \varphi_i\varphi_j \mathrm{d}x \quad \text{and} \quad E_{ij} = \int_\Omega \varphi_{i,x}\varphi_{j,x}\mathrm{d}x,$$

we set $\mathbf{m} = (A+E)\mathbf{u}$. Note that the matrices A and E depend on the mesh, and thus will change when adaptivity is used. We will then distinguish between matrices from different time steps by writing e.g. A^n and A^{n+1}.

Approximating u by u^h as in section 2.2.2, we find

$$\begin{aligned}\mathcal{H}^1_{\mathbf{p}}(\mathbf{m}) = \mathcal{H}^1[m^h] &= \frac{1}{2}\int_\Omega (u^h)^2 + (u^h_x)^2\mathrm{d}x\\ &= \frac{1}{2}\sum_{i,j} u_iu_j\int_\Omega \varphi_i\varphi_j\mathrm{d}x + \frac{1}{2}\sum_{i,j}u_iu_j\int_\Omega \varphi_{i,x}\varphi_{j,x}\mathrm{d}x\\ &= \frac{1}{2}\mathbf{u}^{\mathrm{T}}(A+E)\mathbf{u}\end{aligned}$$

The integrals can be evaluated exactly and efficiently by considering elementwise which basis functions are supported on the element before applying Gaussian quadrature to obtain exact evaluations of the polynomial integrals. We define the matrix $B_1(\mathbf{u})$ by

$$B_1(\mathbf{u})_{ji} = -\frac{1}{3}\sum_{k=0}^{M-1} u_k\left(2\int_\Omega \varphi_i\varphi_{j,x}\varphi_k\mathrm{d}x + \int_\Omega \varphi_i\varphi_j\varphi_{k,x}\mathrm{d}x\right) - \int_\Omega \varphi_i\varphi_{j,x}\mathrm{d}x.$$

An approximation to the gradient of $\mathcal{H}^1$ with respect to m is found by the AVF discrete gradient

$$\begin{aligned}\overline{\nabla}\mathcal{H}^1_{\mathbf{p}}(\mathbf{m}^n,\mathbf{m}^{n+1}) &= (A+E)^{-1}\overline{\nabla}\mathcal{H}^1_{\mathbf{p}}(\mathbf{u}^n,\mathbf{u}^{n+1})\\ &= (A+E)^{-1}\int_0^1 \nabla\mathcal{H}^1_{\mathbf{p}}(\xi\mathbf{u}^n+(1-\xi)\mathbf{u}^{n+1})\mathrm{d}\xi\\ &= (A+E)^{-1}\frac{1}{2}(A+E)\left(\mathbf{u}^n+\mathbf{u}^{n+1}\right)=\frac{1}{2}\left(\mathbf{u}^n+\mathbf{u}^{n+1}\right).\end{aligned}$$

Thus we have the required terms for forming the system (2.4) and applying the adaptive discrete gradient method to it. Corresponding to (2.5), we get the scheme

$$\begin{aligned}(A^{n+1}+E^{n+1})\left(\mathbf{u}^{n+1}-\hat{\mathbf{u}}\right) &= \frac{\hat{\mathbf{u}}^{\mathrm{T}}\left(A^{n+1}+E^{n+1}\right)\hat{\mathbf{u}}-\left(\mathbf{u}^n\right)^{\mathrm{T}}\left(A^n+E^n\right)\mathbf{u}^n}{\left(\hat{\mathbf{u}}+\mathbf{u}^{n+1}\right)^{\mathrm{T}}\left(\hat{\mathbf{u}}+\mathbf{u}^{n+1}\right)}\left(\hat{\mathbf{u}}+\mathbf{u}^{n+1}\right)\\ &\quad+\frac{\Delta t}{2}B_1^{n+1}\left(\frac{\hat{\mathbf{u}}+\mathbf{u}^{n+1}}{2}\right)\left(\hat{\mathbf{u}}+\mathbf{u}^{n+1}\right).\end{aligned}$$

Here we have chosen the skew-symmetric matrix B_1 to be a function of $\hat{\mathbf{u}}$ and $\mathbf{u}^{n+1}$, but could also have chosen e.g. $B_1(\hat{\mathbf{u}})$, resulting in a decreased computational cost at the expense of less precise results. During testing, the basis functions were chosen as piecewise cubic polynomials.

In the same manner we may obtain a scheme that preserves $\mathcal{H}^2[m]$. In this case

$$\begin{aligned}\mathcal{H}^2_{\mathbf{p}}(\mathbf{m}) = \mathcal{H}^2[m^h] &= \frac{1}{2}\int_\Omega (u^h)^2+\frac{1}{3}(u^h)^3\mathrm{d}x\\ &= \frac{1}{2}\sum_{i,j}u_iu_j\int_\Omega \varphi_i\varphi_j\mathrm{d}x+\frac{1}{6}\sum_{i,j,k}u_iu_ju_k\int_\Omega\varphi_i\varphi_j\varphi_k\mathrm{d}x.\end{aligned}$$

and

$$(B_2)_{ji} = -\int_\Omega \varphi_i\varphi_{j,x}\mathrm{d}x+\int_\Omega\varphi_i\varphi_{j,xxx}\mathrm{d}x.$$

Note that the skew-symmetric matrix B_2 is independent of $\mathbf{u}$.

Defining the tensor D by its elements

$$D_{ijk} = \int_\Omega \varphi_i\varphi_j\varphi_k\mathrm{d}x,$$

we get, with the convention of summation over repeated indices, the AVF discrete gradient with respect to $\mathbf{u}$ given by the elements

$$\overline{\nabla}\mathcal{H}^2_{\mathbf{p}}(\mathbf{u}^n,\mathbf{u}^{n+1})_i = \frac{A_{ij}}{2}(u_j^n+u_j^{n+1})+\frac{D_{ijk}}{6}\left(u_j^n(u_k^n+\frac{u_k^{n+1}}{2})+u_j^{n+1}(\frac{u_k^n}{2}+u_k^{n+1})\right)$$

and again the discrete gradient with respect to **m** by

$$\overline{\nabla}\mathcal{H}^2_{\mathbf{p}}(\mathbf{m}^n,\mathbf{m}^{n+1}) = (A+E)^{-1}\overline{\nabla}\mathcal{H}^2_{\mathbf{p}}(\mathbf{u}^n,\mathbf{u}^{n+1}).$$

If we employ integral preserving transfer between the meshes, we get the scheme

$$\mathbf{u}^{n+1} - \hat{\mathbf{u}} = \Delta t(A+E)^{-1}B_2(A+E)^{-1}\overline{\nabla}\mathcal{H}^2_{\mathbf{p}}(\hat{\mathbf{u}},\mathbf{u}^{n+1}),$$

where we note that $S_{\mathbf{p},2} coloneqq (A+E)^{-1}B_2(A+E)^{-1}$ is a skew-symmetric matrix. If non-preserving transfer is used, we need a correction term, as in the $\mathcal{H}^1$ scheme above. The calculation of such a term is straightforward, but we omit it here for reasons of brevity.

To approximate the third derivative in B_2, we need basis functions of at least degree three, and to guarantee skew-symmetry in B_2, these basis functions need to be C^2 on the element boundaries. This is not obtainable with regular nodal FEM basis functions, so we have instead used third order B-spline basis functions as described in [7] during testing.

2.5 Numerical results

To demonstrate the performance of our methods, we have tested them on two one-dimensional simple problems: A soliton solution, and the interaction of two waves. We have tested our $\mathcal{H}^1$- and $\mathcal{H}^2$-preserving schemes on uniform and moving meshes, and compared the results to those obtained using the explicit midpoint method. For the transfer operation between meshes, we have used a piecewise cubic interpolation method in the $\mathcal{H}^1$ preserving scheme, and exact transfer in the $\mathcal{H}^2$ preserving scheme.

2.5.1 Mesh adaptivity

As noted in section 2.3, our methods can be coupled with any adaptive mesh strategy using the rezoning approach. For our numerical experiments, we have used a simple method for r-adaptivity based on the equidistribution principle: Splitting Ω into M intervals, we require that

$$\int_{x_i}^{x_{i+1}} \omega(x)\,\mathrm{d}x = \frac{1}{M}\int_{-L}^{L} \omega(x)\,\mathrm{d}x,$$

where the monitor function ω is a function measuring how densely grid points should lie, based on the value of u. For a general discussion on the choice of an optimal monitor function, see e.g. [3,4]. For the problems we have studied, a generalized solution arc length monitor function proved to yield good results. This is given by

$$\omega(x) = \sqrt{1 + k^2 \left(\frac{\partial u}{\partial x}(x) \right)^2}.$$

For $k = 1$, this is the usual arc length monitor function, in which case the equidistribution principle amounts to requiring that the arc length of u over each interval is equal. In applications, we only have an approximation of u, and hence ω must be approximated as well. We have applied a finite difference approximation and obtained approximately equidistributing grids using de Boor's method as explained in [14, pp. 36-38].

2.5.2 Soliton solution

With $u_0(x) = 3(c-1)\operatorname{sech}^2\left(\frac{1}{2}\sqrt{1-\frac{1}{c}}x\right)$, the exact solution of (2.7)–(2.9) is

$$u(x,t) = 3(c-1)\operatorname{sech}^2\left(\frac{1}{2}\sqrt{1-\frac{1}{c}}\,l(x,t)\right),$$

with $l(x,t) = \min_{j\in\mathbb{Z}} \left|x - ct + 2jL\right|$. This is a soliton solution which travels with a constant speed c in x-direction while maintaining its initial shape.

To evaluate the numerical solutions, we have compared them to the exact solution and calculated errors in shape and phase. The phase error is evaluated as

$$E_n^{\text{phase}} = |ct_n - x^*|,$$

where $x^* = \arg\max_x u_h(x,t_n)$, i.e. the location of the peak of the soliton in the numerical solution. The shape error is given by

$$E_n^{\text{shape}} = \left\| u_h(x,t_n) - u\left(x, \frac{x^*}{c}\right) \right\|,$$

where the peak of the exact solution is translated to match the peak of the numerical solution, and the difference in the shapes of the solitons is calculated.

The results of the numerical tests can be seen in figures 2.1–2.3. Here, M denotes the degrees of freedom used in the spatial approximation and Δt the fixed time step size. DG1 and DG1MM denotes the $\mathcal{H}_{\mathbf{p}}^1$ preserving scheme with fixed, uniform grid and adaptive grid, respectively; similary DG2 and DG2MM denotes the $\mathcal{H}_{\mathbf{p}}^2$ preserving scheme with uniform and adaptive grids.

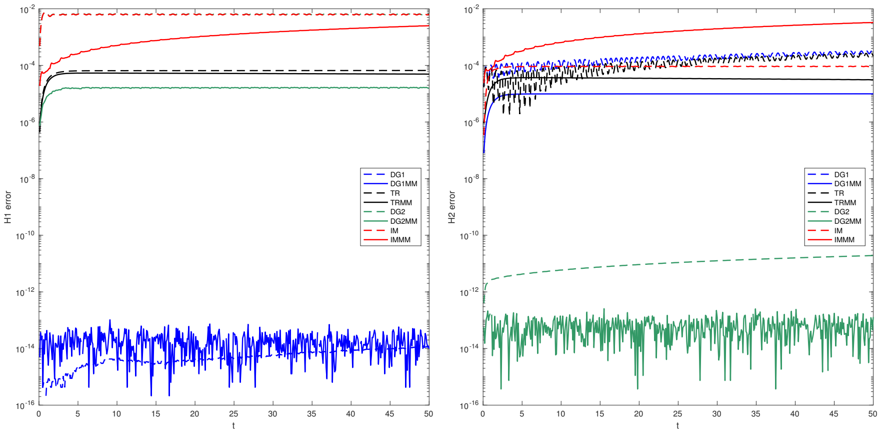


Figure 2.1: The soliton problem. Relative error in the approximated Hamiltonians $\mathcal{H}_{\mathbf{p}}^1$ (left) and $\mathcal{H}_{\mathbf{p}}^2$ (right) plotted as a function of time $t \in [0,50]$. $c = 3, L = 200, \Delta t = 0.1$, $M = 200$.

In Figure 2.1 we see the relative errors in $\mathcal{H}_{\mathbf{p}}^1$ and $\mathcal{H}_{\mathbf{p}}^2$. The DG1 and DG1MM schemes are compared to schemes using the same 3rd order nodal basis functions, but the trapezoidal rule for time-stepping, denoted by TR and TRMM. Likewise, the DG2 and DG2MM schemes are compared to the IM and IMMM schemes, using B-spline basis functions and the implicit midpoint method for discretization in time. The error in $\mathcal{H}_{\mathbf{p}}^1$ is very small for the DG1 and DG1MM schemes, as expected. Also the error in $\mathcal{H}_{\mathbf{p}}^2$ is very small for the DG2 and DG2MM schemes. The order of the error is not machine precision, but is instead dictated by the precision with which the nonlinear equations in each time step is solved. We can also see that while the TR and IM schemes, with and without moving meshes, have poor conservation properties, the moving mesh DG schemes seem to preserve quite well even the integrals they are not designed to preserve.

In figures 2.2 and 2.3 we see the phase and shape errors, of our methods compared to non-moving mesh methods and non-preserving methods, respectively. The advantage of using moving meshes is clear, especially for the $\mathcal{H}_{\mathbf{p}}^2$ preserving schemes. The usefulness on integral preservation is ambiguous in this case. It seems that what we gain in precision in phase, we lose in precision in shape, and vice versa.

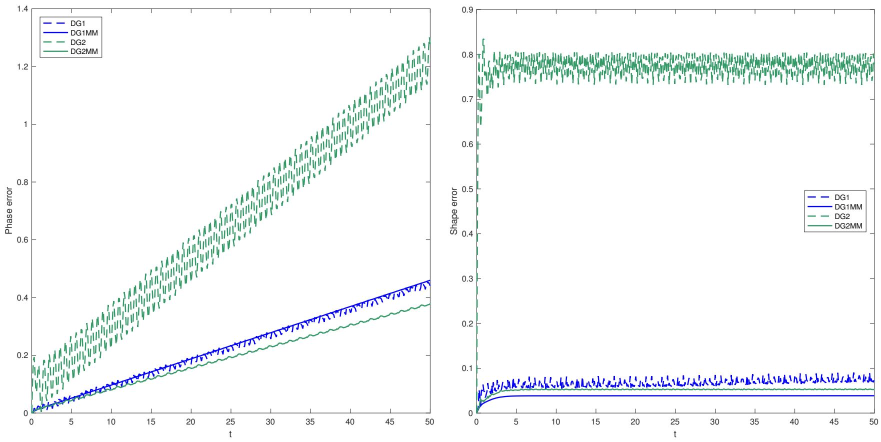


Figure 2.2: The soliton problem. Phase error (left) and shape error (right) as a function of time. $c = 3, L = 200, \Delta t = 0.1,\ M = 200$.

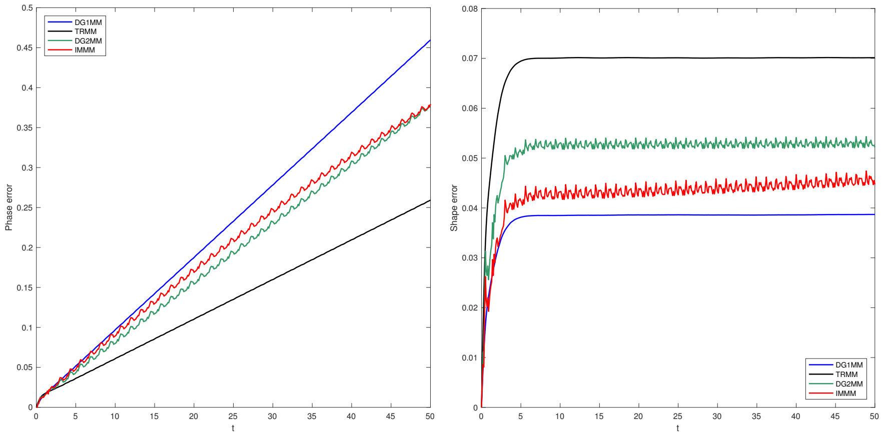


Figure 2.3: The soliton problem. Phase error (left) and shape error (right) as a function of time. $c = 3, L = 200, \Delta t = 0.1,\ M = 200$.

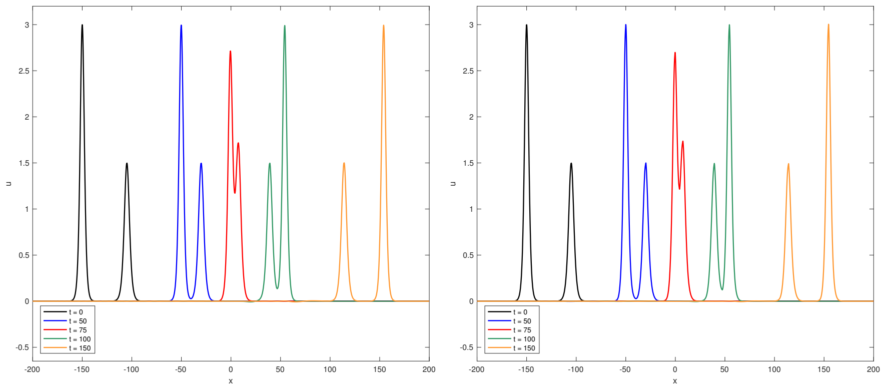


Figure 2.4: The interacting waves problem. Solutions at $t = \{0, 50, 75, 100, 150\}$ found by DG1MM (left) and DG2MM (right). $x_r = 150, x_s = 105, c_r = 2, c_s = 1.5, L = 200, \Delta t = 0.1,\ M = 1000$.

2.5.3 A small wave overtaken by a large one

A typical test problem for the BBM equation is the interaction between two solitary waves. With an initial condition

$$u_0(x) = 3(c_r - 1)\,\mathrm{sech}^2\left(\sqrt{1 - \frac{1}{c_r}}\,\frac{x - x_r}{2}\right) + 3(c_s - 1)\,\mathrm{sech}^2\left(\sqrt{1 - \frac{1}{c_s}}\,\frac{x - x_s}{2}\right),$$

one wave will eventually be overtaken by the other as long as $c_r \neq c_s$, i.e. if one wave is larger than the other. There is no available analytical solution for this problem. The two waves are not solitons, as the amplitudes will change a bit after the waves have interacted [9].

Solutions obtained by solving the problem with our two energy preserving schemes, giving very similar results, are plotted in Figure 2.4. Also, to illustrate the mesh adaptivity, we have included a plot of the mesh trajectories in Figure 2.6. Each line represents the trajectory of one mesh point in time, and we can see that the mesh points cluster nicely around the edges of the waves as they move.

To illustrate the performance of our methods, we have in Figure 2.5 compared solutions obtained by using the $\mathcal{H}^2_{\mathbf{p}}$-preserving moving mesh method with the solutions obtained by using a fourth order Runge–Kutta method on a static mesh, with the same, and quite few, degrees of freedom. The DG2MM solution is visibly closer to the solutions in Figure 2.4. The non-preserving RK scheme does a worse job of preserving the amplitude and speed of the waves compared to the DG2MM scheme, and we observe unwanted oscillations.

In Figure 2.7 we have plotted the Hamiltonian errors for this problem.

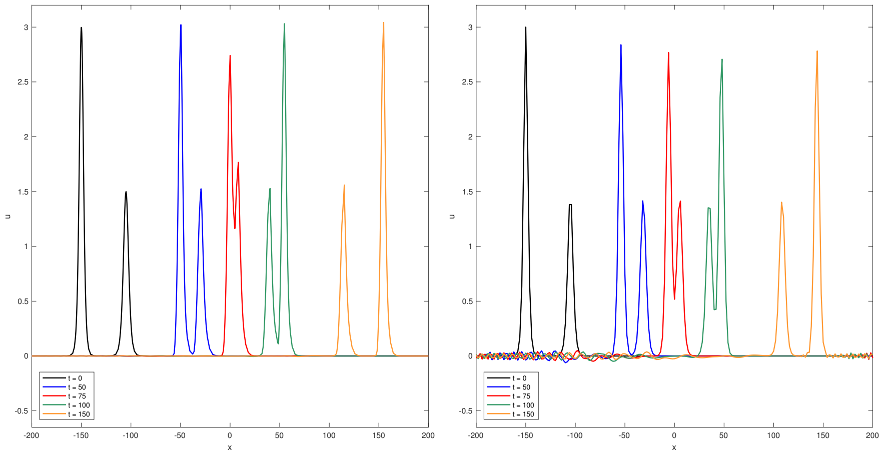


Figure 2.5: The interacting waves problem. Solutions at $t = \{0, 50, 75, 100, 150\}$ found by DG2MM (left) and RK (right). $x_r = 150, x_s = 105, c_r = 2, c_s = 1.5, L = 200, \Delta t = 0.1, M = 200$.

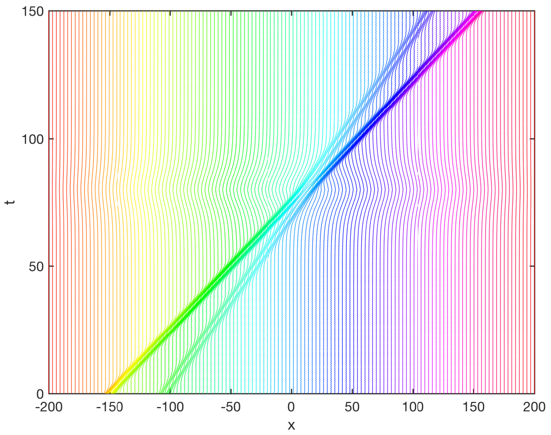


Figure 2.6: Mesh point trajectories in time. Each line represents one mesh point.

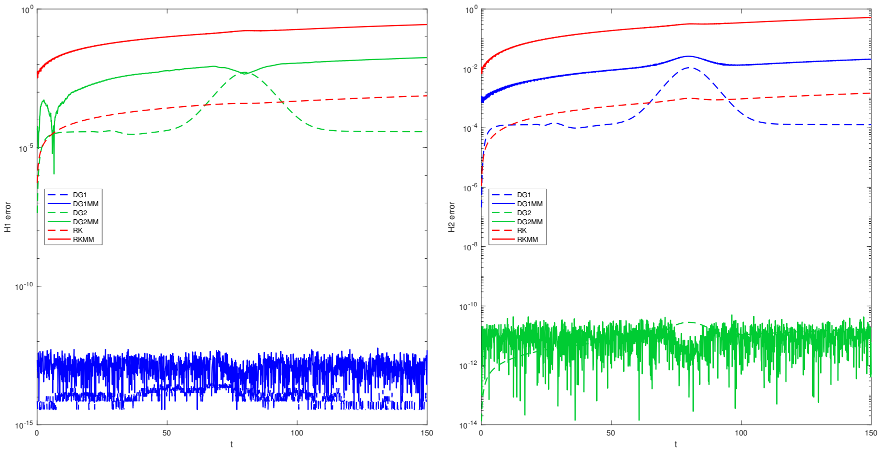


Figure 2.7: The interacting waves problem. Error in the approximated Hamiltonians $\mathcal{H}_{\mathbf{p}}^{1}$ (left) and $\mathcal{H}_{\mathbf{p}}^{2}$ (right) plotted as a function of time $t \in [0, 150]$. $x_r = 150, x_s = 105, c_r = 2, c_s = 1.5, L = 200, \Delta t = 0.1$, $M = 1000$.

Again we see that the energy preserving schemes preserve both Hamiltonians better than the Runge–Kutta scheme, but we do also observe that the DG1 scheme preserves $\mathcal{H}_{\mathbf{p}}^{2}$ better than the DG1MM scheme, and vice versa for the DG2 and DG2MM schemes. Note also that an increase in the errors can be observed when the two waves interact, but that this increase is temporary.

2.6 Conclusions

In this paper, we have presented energy preserving schemes for a class of PDEs, first on general fixed meshes, and then on adaptive meshes. These schemes are then applied to the BBM equation, for which discrete schemes preserving two of the Hamiltonians of the problem are explicitly given.

Numerical experiments are performed, using the energy preserving moving mesh schemes on two different BBM problems: a soliton solution, and two waves interacting. Plots of the phase and shape errors illustrate how, for the given parameters, the usage of moving meshes gives improved accuracy, while the integral preservation gives comparable results to existing methods, without yielding a categorical improvement. We will remark, however, that in many cases, the preservation of a quantity such as one of the Hamiltonians in itself may be a desired property of a numerical scheme. For the two wave interaction problem, we do not have an analytical solution to compare to, but plots of the solution indicate that our schemes perform well compared to a Runge–Kutta scheme.

Although the numerical examples presented here are simple one-dimensional problems, the adaptive discrete gradient methods should also be applicable for multi-dimensional problems. This could be an interesting direction for further work, since the advantages of adaptive meshes are typically more evident when increasing the number of dimensions.

Bibliography

[1] I. Babuška and B. Guo. The h, p and h-p version of the finite element method; basis theory and applications. *Adv. Eng. Softw.*, 15:159–174, 1992.

[2] T. B. Benjamin, J. L. Bona, and J. J. Mahony. Model equations for long waves in nonlinear dispersive systems. *Philos. T. R. Soc. A.*, 272(1220):47–78, 1972.

[3] J. Blom and J. Verwer. On the use of the arclength and curvature monitor in a moving-grid method which is based on the method of lines. Technical report, NM-N8902, CWI, Amsterdam, 1989.

[4] C. J. Budd, W. Huang, and R. D. Russell. Adaptivity with moving grids. *Acta Numer.*, 18:111–241, 2009.

[5] 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(20):6770–6789, 2012.

[6] D. Cohen and X. Raynaud. Geometric finite difference schemes for the generalized hyperelastic-rod wave equation. *J. Comput. Appl. Math.*, 235(8):1925–1940, 2011.

[7] J. A. Cottrell, T. J. Hughes, and Y. Bazilevs. *Isogeometric Analysis*. Wiley, 2009.

[8] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Ann.*, 100(1):32–74, 1928.

[9] W. Craig, P. Guyenne, J. Hammack, D. Henderson, and C. Sulem. Solitary water wave interactions. *Phys. Fluids*, 18(5):057106, 2006.

[10] S. Eidnes, B. Owren, and T. Ringholm. Adaptive energy preserving methods for partial differential equations. *Adv. Comput. Math.*, 44(3):815–839, 2018.

[11] D. Furihata and T. Matsuo. *Discrete variational derivative method*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011.

[12] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[13] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[14] W. Huang and R. Russell. *Adaptive moving mesh methods*, volume 174 of *Springer Series in Applied Mathematical Sciences*. Springer-Verlag, New York, 2010.

[15] S. Li and L. Vu-Quoc. Finite difference calculus invariant structure of a class of algorithms for the nonlinear Klein-Gordon equation. *SIAM J. Numer. Anal.*, 32(6):1839–1875, 1995.

[16] C. Lu, W. Huang, and J. Qiu. An adaptive moving mesh finite element solution of the regularized long wave equation. *J. Sci. Comput.*, 74(1):122–144, 2018.

[17] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *Philos. T. R. Soc. A*, 357(1754):1021–1045, 1999.

[18] Y. Miyatake and T. Matsuo. A note on the adaptive conservative/dissipative discretization for evolutionary partial differential equations. *J. Comput. Appl. Math.*, 274:79–87, 2015.

[19] D. Peregrine. Calculations of the development of an undular bore. *J. Fluid Mech.*, 25(02):321–330, 1966.

[20] T.-C. Wang and L.-M. Zhang. New conservative schemes for regularized long wave equation. *Numer. Math. Chin.*, 15(4):348–356, 2006.

[21] T. Yaguchi, T. Matsuo, and M. Sugihara. An extension of the discrete variational method to nonuniform grids. *J. Comput. Phys.*, 229(11):4382–4423, 2010.

[22] T. Yaguchi, T. Matsuo, and M. Sugihara. The discrete variational derivative method based on discrete differential forms. *J. Comput. Phys.*, 231(10):3963–3986, 2012.

Energy preserving methods on Riemannian manifolds

Elena Celledoni, Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm

Published in Mathematics of Computation 89 (322), pp. 699–716 (2020)

Energy preserving methods on Riemannian manifolds

Abstract. The energy preserving discrete gradient methods are generalized to finite-dimensional Riemannian manifolds by definition of a discrete approximation to the Riemannian gradient, a retraction, and a coordinate center function. The resulting schemes are formulated only in terms of these three objects and do not otherwise depend on a particular choice of coordinates or embedding of the manifold in a Euclidean space. Generalizations of well-known discrete gradient methods, such as the average vector field method and the Itoh–Abe method, are obtained. It is shown how methods of higher order can be constructed via a collocation-like approach. Local and global error bounds are derived in terms of the Riemannian distance function and the Levi-Civita connection. Numerical results are presented, for problems on the two-sphere, the paraboloid and the Stiefel manifold.

3.1 Introduction

A first integral of an ordinary differential equation (ODE) is a scalar-valued function on the phase space of the ODE that is preserved along solutions. The potential benefit of using numerical methods that preserve one or more such invariants is well-documented, and several energy-preserving methods have been developed in recent years. Among these are the discrete gradient methods, which were introduced for use in Euclidean spaces in [10], see also [23]. These methods are based on the idea of expressing the ODE using a skew-symmetric operator and the gradient of the first integral, and then creating a discrete counterpart to this in such a way that the numerical scheme preserves the first integral.

For manifolds in general, one can use the same schemes expressed in local coordinates. A drawback is that the numerical approximation will typically depend on the particular choice of coordinates and also on the strategy used for transition between coordinate charts. Another alternative is to use a global embedding of the manifold into a larger Euclidean space, but then it typically happens that the numerical solution deviates from the manifold. Even if the situation can be amended by using projection, it may not be desirable that the computed approximation depends on the particular embedding chosen. Crouch and Grossmann [7] and Munthe-Kaas [25, 26] introduced different ways of extending existing Runge–Kutta methods to a large class of differentiable manifolds. Both these approaches are generally classified as Lie group integrators, see [14] or the more recent [4] for a survey of this class of methods. They can also both be formulated abstractly by means of a post-Lie structure which consists of a Lie algebra with a flat connection of constant torsion, see e.g. [27].

In the present paper we shall state the methods in a slightly different context, using the notion of a Riemannian manifold. It is then natural to make use of the Levi-Civita connection, which in contrast to the post-Lie setting is torsion-free, and which in general has a non-zero curvature. For our purposes it is also an advantage that the Riemannian metric provides an intrinsic definition of the gradient. Taking an approach more in line with this, Leimkuhler and Patrick [19] considered mechanical systems on the cotangent bundle of a Riemannian manifold and succeeded in generalising the classical leap-frog scheme to a symplectic integrator on Riemannian manifolds.

Some classical numerical methods in Euclidean spaces preserve certain classes of invariants; for instance, symplectic Runge–Kutta methods preserve all quadratic invariants. This can be useful when there is a natural way of embedding a manifold into a linear space by using constraints that are expressed by means of such invariants. An example is the 2-sphere which can be embedded in $\mathbb{R}^3$ by adding the constraint that these vectors should have unit length. The classical midpoint rule will automatically ensure that the numerical approximations remain on the sphere as it preserves all quadratic invariants. In general, however, the invariants preserved by these methods are expressed in terms of coordinates. Hence the preservation property of the method may be lost under coordinate changes if the invariant is no longer quadratic. In [5], a generalization of the discrete gradient method to differential equations on Lie groups and a broad class of manifolds was presented. Here we develop this further by introducing a Riemannian structure that can be used to provide an intrinsic definition of the gradient and a means to measure numerical errors.

The structure of this paper is as follows: In section 2, we formulate the problem to be solved and introduce discrete Riemannian gradient methods, as well as presenting some particular examples with special attention to a generalization of the Itoh–Abe discrete gradient. We also briefly discuss the Euclidean setting as a special choice of manifold and show how the standard discrete gradient methods are recovered in this case. In the third section, we consider higher order energy preserving methods based on generalization of a collocation strategy introduced by Hairer [12] to Riemannian manifolds. We present some error analysis in section 4, and show numerical results in section 5, where the methods are applied to models of a body moving on the two-sphere and on the paraboloid, and on a system on the Stiefel manifold.

3.2 Energy preservation on Riemannian manifolds

Consider an initial value problem on the finite-dimensional Riemannian manifold (M,g),

$$\dot{u} = F(u), \quad u(0) = u^0 \in M, \tag{3.1}$$

where F is a smooth vector field, $u^0 \in M$ is the initial value and g is a Riemannian metric. We denote by $\mathcal{F}(M)$ the space of smooth functions on M. The set of smooth vector fields and differential one-forms are denoted $\Gamma(TM)$ and $\Gamma(T^*M)$ respectively, and for the duality pairing between these two spaces we use the angle brackets $\langle\cdot,\cdot\rangle$.

A first integral associated with a vector field $F \in \Gamma(TM)$ is a function $H \in \mathcal{F}(M)$ such that $\langle \mathrm{d}H, F\rangle$ vanishes identically on M. First integrals are preserved along solutions of (3.1), since

$$\frac{\mathrm{d}}{\mathrm{d}t} H(u(t)) = \big\langle \mathrm{d}H(u(t)), \dot{u}(t)\big\rangle = \big\langle \mathrm{d}H(u(t)), F(u(t))\big\rangle = 0.$$

3.2.1 Preliminaries

The fact that a vector field F has a first integral H is closely related to the existence of a tensor field $\Omega \in \Gamma(TM \otimes T^*M) =: \Gamma(\mathcal{T}_1^1 M)$, skew-symmetric with respect to the metric g, such that

$$F(u) = \Omega(u)\,\mathrm{grad} H(u), \tag{3.2}$$

where $\mathrm{grad} H \in \Gamma(TM)$ is the Riemannian gradient, the unique vector field satisfying $\langle \mathrm{d}H, \cdot\rangle = g(\mathrm{grad} H, \cdot)$. Any ODE (3.1) where F is of this form preserves H, since

$$\frac{\mathrm{d}}{\mathrm{d}t} H(u) = \big\langle \mathrm{d}H(u), \dot{u}\big\rangle = \big\langle \mathrm{d}H(u), \Omega\,\mathrm{grad} H(u)\big\rangle = g(\mathrm{grad} H(u), \Omega\,\mathrm{grad} H(u)) = 0.$$

A converse result is detailed in the following proposition.

Proposition 3.1. Any system (3.1) with a first integral H can be written with an F of the form (3.2). The skew tensor field Ω can be chosen so as to be bounded near every nondegenerate critical point of H.

Proof. Similar to the proof of Proposition 2.1 in [23], we can write an explicit expression for a possible choice of Ω,

$$\Omega y = \frac{g(\mathrm{grad} H, y)\, F - g(F, y)\,\mathrm{grad} H}{g(\mathrm{grad} H, \mathrm{grad} H)}. \tag{3.3}$$

Clearly, $g(y, \Omega y) = 0$ for all y. Since H is a first integral, $g(F, \mathrm{grad} H) = \langle \mathrm{d}H, F\rangle = 0$, so $\Omega\,\mathrm{grad} H = F$. For a proof that Ω is bounded near nondegenerate critical points, see [23]. □

In fact, such a tensor field Ω often arises naturally from a two-form ω through the assignment $\Omega y = \omega(\cdot, y)^\sharp$. A well-known example is when ω is a symplectic two-form. Note that Ω is not necessarily unique.

Retractions, viewed as maps from TM to M, will play an important role in the methods we discuss here. Their formal definition can be found e.g. in [1]:

Definition 3.1. Let ϕ be a smooth map defined on a neighborhood of M in TM and let ϕ_p denote the restriction of ϕ to the tangent space T_pM at $p \in M$, with 0_p being the zero-vector in T_pM. Then ϕ is a *retraction* if it satisfies the conditions

1. ϕ_p is defined in an open ball $B_{r_p}(0_p) \subset T_pM$ of radius r_p about 0_p,
2. $\phi_p(x) = p$ if and only if $x = 0_p$,
3. $D\phi_p\big|_{0_p} = \mathrm{Id}_{T_pM}$.

A canonical example of a retraction on (M, g) is obtained via the Riemannian exponential, setting $\phi_p(x) = \exp_p(x)$, i.e., following along the geodesic emanating from p in the direction x. The Riemannian exponential may be more computationally expensive to evaluate than other retractions, but its geometric position in the Riemannian framework could provide for an informative error analysis.

3.2.2 The discrete Riemannian gradient method

We adapt the discrete gradients in Euclidean space to discrete Riemannian gradients (DRG) on (M, g) by means of a retraction map ϕ and a center point function c.

Definition 3.2. A discrete Riemannian gradient is a triple $(\overline{\mathrm{grad}}, \phi, c)$[1] where

1. $c : M \times M \to M$ is a continuous map such that $c(u,u) = u$ for all $u \in M$,
2. $\overline{\mathrm{grad}} : \mathcal{F}(M) \to \Gamma(c^* TM)$,
3. $\phi : TM \to M$ is a retraction,

such that for all $H \in \mathcal{F}(M)$, $u \in M$, $v \in M$, $c = c(u,v) \in M$,

$$H(v) - H(u) = g(\overline{\mathrm{grad}}H(u,v), \phi_c^{-1}(v) - \phi_c^{-1}(u)), \tag{3.4}$$

$$\overline{\mathrm{grad}}H(u,u) = \mathrm{grad}H(u). \tag{3.5}$$

[1] To avoid cluttered notation we will just write $\overline{\mathrm{grad}}$ for the triple $(\overline{\mathrm{grad}}, \phi, c)$ in the sequel.

The DRG $\overline{\mathrm{grad}}H$ is a continuous section of the pullback bundle c^*TM, meaning that $\pi \circ \overline{\mathrm{grad}}H = c$, where $\pi : TM \to M$ is the natural projection. We also need to define an approximation to be used for the tensor field $\Omega \in \Gamma(\mathcal{T}_1^1 M)$. To this end we let $\overline{\Omega} \in \Gamma(c^*\mathcal{T}_1^1 M)$ be a continuous skew-symmetric tensor field such that

$$\overline{\Omega}(u,u) = \Omega(u) \quad \forall u \in M.$$

Inspired by [3, 5], we propose the scheme

$$u^{k+1} = \phi_{c^k}(W(u^k, u^{k+1})), \quad c^k = c(u^k, u^{k+1}) \tag{3.6}$$

$$W(u^k, u^{k+1}) = \phi_{c^k}^{-1}(u^k) + h\overline{\Omega}(u^k, u^{k+1})\,\overline{\mathrm{grad}}H(u^k, u^{k+1}), \tag{3.7}$$

where h is the step size. The scheme (3.6)–(3.7) preserves the invariant H, since

$$\begin{aligned} H(u^{k+1}) - H(u^k) &= g(\overline{\mathrm{grad}}H(u^k, u^{k+1}), \phi_{c^k}^{-1}(u^{k+1}) - \phi_{c^k}^{-1}(u^k)) \\ &= g(\overline{\mathrm{grad}}H(u^k, u^{k+1}), h\overline{\Omega}(u^k, u^{k+1})\,\overline{\mathrm{grad}}H(u^k, u^{k+1})) = 0. \end{aligned}$$

Here and in the following we adopt the shorthand notation $c = c(u,v)$ as long as it is obvious what the arguments of c are.

The Average Vector Field (AVF) method has been studied extensively in the literature; some early references are [13, 23, 28]. This is a discrete gradient method, and we propose a corresponding DRG satisfying (3.4)-(3.5) as follows:

$$\overline{\mathrm{grad}}_{\mathrm{AVF}}H(u,v) = \int_0^1 (D_{\gamma_\xi}\phi_c)^{\mathrm{T}}\,\mathrm{grad}H(\phi_c(\gamma_\xi))\,\mathrm{d}\xi, \tag{3.8}$$

where $\gamma_\xi = (1-\xi)\phi_c^{-1}(u) + \xi\phi_c^{-1}(v)$, and $(D_x\phi_c)^{\mathrm{T}} : T_{\phi_c(x)}M \to T_cM$ is the unique operator satisfying

$$g((D_x\phi_c)^{\mathrm{T}}a, b) = g(a, D_x\phi_c\, b), \quad \forall x, b \in T_cM, \quad a \in T_{\phi_c(x)}M.$$

Furthermore, we have the generalization of Gonzalez' midpoint discrete gradient [10],

$$\begin{aligned} \overline{\mathrm{grad}}_{\mathrm{MP}}H(u,v) = &\,\mathrm{grad}H(c(u,v)) \\ &+ \frac{H(v) - H(u) - g(\mathrm{grad}H(c(u,v)), \eta)}{g(\eta,\eta)}\eta \end{aligned} \tag{3.9}$$

where $\eta = \phi_c^{-1}(v) - \phi_c^{-1}(u)$.

Note that both these DRGs involve the gradient of the first integral. This may be a disadvantage if H is non-smooth or if its gradient is expensive to compute. Also, the implicit nature of the schemes requires the solution of an n-dimensional nonlinear system of equations at each time step. An alternative is to consider the Itoh–Abe discrete gradient [15], also called the coordinate increment discrete gradient [23], which in certain cases requires only the solution of n decoupled scalar equations. We now present a generalization of the Itoh–Abe discrete gradient to finite-dimensional Riemannian manifolds.

3.2.3 Itoh–Abe discrete Riemannian gradient

Definition 3.3. For any tangent space T_cM one can choose a basis $\{E_1,...,E_n\}$ composed of tangent vectors E_i, $i = 1,...,n$, orthonormal with respect to the Riemannian metric g. Then, given $u, v \in M$, there exists a unique $\{\alpha_i\}_{i=1}^n$ so that

$$\phi_c^{-1}(v) - \phi_c^{-1}(u) = \sum_{i=1}^n \alpha_i E_i.$$

The *Itoh–Abe DRG* of the first integral H is then given by

$$\overline{\mathrm{grad}}_{\mathrm{IA}} H(u,v) = \sum_{j=1}^n a_j E_j, \tag{3.10}$$

where

$$a_j = \begin{cases} \dfrac{H(w_j) - H(w_{j-1})}{\alpha_j} & \text{if } \alpha_j \neq 0, \\ g(\mathrm{grad}H(w_{j-1}), D\phi_c(\eta_{j-1})E_j) & \text{if } \alpha_j = 0, \end{cases}$$

$$w_j = \phi_c(\eta_j), \quad \eta_j = \phi_c^{-1}(u) + \sum_{i=1}^j \alpha_i E_i.$$

We refer to [3] for proof that this is indeed a DRG satisfying (3.4)-(3.5).

3.2.4 Euclidean setting

Let $M = V$ be an $\mathbb{R}$-linear space, and let g be the Euclidean inner product, $g(x,y) = x^{\mathrm{T}}y$. The operator Ω is a solution dependent skew-symmetric $n \times n$ matrix $\Omega(u)$. For any $u \in V$, we have $T_uV \equiv V$. The retraction $\phi : V \to V$ is defined as $\phi_p(x) = p + x$, the Riemannian exponential on V, so that $\phi_c^{-1}(v) - \phi_c^{-1}(u) = v - u$. The gradient grad$H$ is an n-vector whose ith component is $\frac{\partial H}{\partial u_i}$, and the definition of the discrete Riemannian gradient coincides with the standard discrete gradient, since (3.4) now reads

$$H(v) - H(u) = \overline{\mathrm{grad}}(u,v)^{\mathrm{T}}(v-u).$$

Furthermore, (3.6)-(3.7) simply becomes the discrete gradient method introduced in [10], given by the scheme

$$u^{k+1} - u^k = h\overline{\Omega}(u^k, u^{k+1})\overline{\mathrm{grad}}H(u^k, u^{k+1}), \tag{3.11}$$

where $\overline{\Omega}$ is a skew-symmetric matrix approximating Ω. Typical choices are $\overline{\Omega}(u^k, u^{k+1}) = \Omega(u^k)$, or $\overline{\Omega}(u^k, u^{k+1}) = \Omega((u^{k+1} + u^k)/2)$ if one seeks a symmetric method.

The DRGs (3.8) and (3.9) become the standard AVF and midpoint discrete gradients in this case. For the Itoh–Abe DRG, the practical choice for the orthogonal basis would be the set of unit vectors, $\{e_1, ..., e_n\}$, so that $\alpha_i = v_i - u_i$, and we get (3.10) with

$$a_j = \begin{cases} \dfrac{H(w_j) - H(w_{j-1})}{v_j - u_j} & \text{if } u_j \neq v_j, \\ \frac{\partial H}{\partial u_j}(w_{j-1}) & \text{if } u_j = v_j, \end{cases}$$
$$w_j = \textstyle\sum_{i=1}^{j} v_i e_i + \sum_{i=j+1}^{n} u_i e_i,$$

which is a reformulation of the Itoh–Abe discrete gradient as it is given in [15], [23] and the literature otherwise.

3.2.5 Lie group setting

Consider the case where M is a Lie group, $M = G$, equipped with a right-invariant Riemannian metric g. The methods described in reference [5] can be seen as a special case of the methods presented in the current paper, with the retraction map chosen to be the Lie group exponential, see [5] for details. In the special case when the Riemannian metric is bi-invariant (and the exponential map of the Lie group setting coincides with the Riemannian exponential [24]) the methods of [5] are an example of the methods presented here, implemented using normal coordinates (see [17, p. 76]).

3.3 Methods of higher order

In the Euclidean setting, a strategy to obtain energy preserving methods of higher order was presented in [2] and later in [12], see also [6]. This technique is generalized to a Lie group setting in [5]. We will here formulate these methods in the context of Riemannian manifolds.

3.3.1 Energy-preserving collocation-like methods on Riemannian manifolds

Let $\mathsf{c}_1, ..., \mathsf{c}_s$ be distinct real numbers, where s is the order of the collocation polynomial specified below. Consider the Lagrange basis polynomials,

$$l_i(\xi) = \prod_{j=1, j\neq i}^{s} \frac{\xi - \mathsf{c}_j}{\mathsf{c}_i - \mathsf{c}_j}, \quad \text{and let} \quad b_i := \int_0^1 l_i(\xi)\, d\xi. \tag{3.12}$$

We assume that $\mathsf{c}_1, \ldots, \mathsf{c}_s$ are such that $b_i \neq 0$ for all i. A step of the energy-preserving collocation-like method, starting at $u^0 \in M$, is defined via a polynomial $\sigma : \mathbb{R} \to T_c M$ of degree s satisfying

$$\sigma(0) = \phi_c^{-1}(u^0), \tag{3.13}$$

$$\frac{d}{d\xi}\sigma(\xi h)\Big|_{\xi=\mathrm{c}_j} = D_{U_j}\phi_c^{-1}\left(\Omega_j \mathrm{grad}_j H\right), \quad U_j := \phi_c\left(\sigma(\mathrm{c}_j h)\right) \tag{3.14}$$

$$u^1 := \phi_c\left(\sigma(h)\right), \tag{3.15}$$

where

$$\mathrm{grad}_j H := \int_0^1 \frac{l_j(\xi)}{b_j}\left(D_{U_j}\phi_c^{-1}\right)^{\mathrm{T}}(D_{\sigma(\xi h)}\phi_c)^{\mathrm{T}}\,\mathrm{grad}H\left(\phi_c(\sigma(\xi h))\right)\mathrm{d}\xi,$$

and $\Omega_j := \Omega(U_j)$. Notice that with $s = 1$ and independently on the choice of c_1, we reproduce the DRG method (3.6)-(3.7) with the AVF DRG (3.8).

Using Lagrange interpolation and (3.14), the derivative of $\sigma(\xi h)$ at every point ξh is

$$\frac{d}{d\xi}\sigma(\xi h) = \sum_{j=1}^{s} l_j(\xi) D_{U_j}\phi_c^{-1}\left(\Omega_j \mathrm{grad}_j H\right), \tag{3.16}$$

from which by integrating we get

$$\sigma(\tau h) = \phi_c^{-1}(u_0) + h\sum_{j=1}^{s}\int_0^{\tau} l_j(\xi)\,\mathrm{d}\xi\, D_{U_j}\phi_c^{-1}\left(\Omega_j \mathrm{grad}_j H\right).$$

The defined method is energy preserving, which we see by using

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\left(\phi_c(\sigma(\xi h))\right) = D_{\sigma(\xi h)}\phi_c\left(\frac{\mathrm{d}}{\mathrm{d}\xi}\sigma(\xi h)\right),$$

and (3.16) to get

$$\begin{aligned}
H(u^1) - H(u^0) &= \int_0^1 g\left(\mathrm{grad}H\left(\phi_c(\sigma(\xi h))\right), \frac{\mathrm{d}}{\mathrm{d}\xi}\phi_c(\sigma(\xi h))\right)\mathrm{d}\xi \\
&= \int_0^1 g\left(\mathrm{grad}H\left(\phi_c(\sigma(\xi h))\right), D_{\sigma(\xi h)}\phi_c\left(\sum_{j=1}^{s} l_j(\xi)\, D_{U_j}\phi_c^{-1}\left(\Omega_j \mathrm{grad}_j H\right)\right)\right)\mathrm{d}\xi \\
&= \int_0^1 g\left(\left(D_{\sigma(\xi h)}\phi_c\right)^{\mathrm{T}}\mathrm{grad}H\left(\phi_c(\sigma(\xi h))\right), \sum_{j=1}^{s} l_j(\xi)\, D_{U_j}\phi_c^{-1}\left(\Omega_j \mathrm{grad}_j H\right)\right)\mathrm{d}\xi \\
&= \sum_{j=1}^{s} b_j g\left(\int_0^1 \frac{l_j(\xi)}{b_j}\left(D_{U_j}\phi_c^{-1}\right)^{\mathrm{T}}\left(D_{\sigma(\xi h)}\phi_c\right)^{\mathrm{T}}\mathrm{grad}H\left(\phi_c(\sigma(\xi h))\right)\mathrm{d}\xi, \Omega_j \mathrm{grad}_j H\right) \\
&= \sum_{j=1}^{s} b_j g\left(\mathrm{grad}_j H, \Omega_j\,\mathrm{grad}_j H\right) = 0,
\end{aligned}$$

and hence repeated use of (3.13)-(3.15) ensures $H(u^k) = H(u^0)$ for all $k \in \mathbb{N}$.

3.3.2 Higher order extensions of the Itoh–Abe DRG method

From the Itoh–Abe DRG one can get a new DRG, also satisfying (3.4), by

$$\overline{\operatorname{grad}}_{\mathrm{SIA}}H(u,v)=\frac{1}{2}\left(\overline{\operatorname{grad}}_{\mathrm{IA}}H(u,v)+\overline{\operatorname{grad}}_{\mathrm{IA}}H(v,u)\right). \tag{3.17}$$

We call this the *symmetrized Itoh–Abe DRG*. Note that we need the base point c to be the same in the evaluation of $\overline{\operatorname{grad}}_{\mathrm{IA}}H(u,v)$ and $\overline{\operatorname{grad}}_{\mathrm{IA}}H(v,u)$. When $c(u,v)=c(v,u)$ and $\overline{\Omega}_{(u,v)}=\overline{\Omega}_{(v,u)}$, we get a symmetric DRG method (3.6)-(3.7), which is therefore of second order.

Alternatively, one can get a symmetric 2-stage method by a composition of the Itoh–Abe DRG method and its adjoint. Furthermore, one can get energy preserving methods of any order using a composition strategy. To ensure symmetry of an s-stage composition method, one needs $c_i(u,v)=c_{s+1-i}(v,u)$ for different center points c_i belonging to each stage and, similarly, $\overline{\Omega}_i(u,v)=\overline{\Omega}_{s+1-i}(v,u)$.

3.4 Error analysis

3.4.1 Local error

In this section, $\varphi_t(u)$ is the t-flow of the ODE vector field F. The most standard discrete gradient methods have a low or moderate order of convergence, and this is also the case for the DRG methods unless special care is taken in designing $\overline{\Omega}$ and $\overline{\operatorname{grad}}H$. We shall not pursue this approach here, but refer to the collocation-like methods if high order of accuracy is required. We shall see, however, that the methods designed here are consistent and can be made symmetric. Analysis of the local error can be done in local coordinates, assuming that the step size is always chosen sufficiently small, so that within a fixed step, $u^k, u^{k+1}, c(u^k, u^{k+1})$ and the exact local solution $u(t_{k+1})$ all belong to the same given coordinate chart. From the definition (3.6)-(3.7) it follows immediately that the representation of $u^{k+1}(h)$ satisfies $u^{k+1}(0)=u^k$ and $\frac{d}{dh}u^{k+1}(0)=F(u^k)$. Then, by equivalence of local coordinate norms and the Riemannian distance, we may conclude that the local error in DRG methods satisfies

$$d(u^{k+1},\varphi_h(u^k))\leq Ch^2.$$

Similar to what was also observed in [5], the DRG methods (3.6)-(3.7) are symmetric whenever $\overline{\operatorname{grad}}H(u,v)=\overline{\operatorname{grad}}H(v,u)$, $\overline{\Omega}(u,v)=\overline{\Omega}(v,u)$, and $c(u,v)=c(v,u)$ for all $u,v\in M$. In that case we obtain an error bound for the local error of the form $d(u^{k+1},\varphi_h(u^k))\leq Ch^3$.

The collocation-like methods of section 3.3 have associated nodes $\{\mathrm{c}_i\}_{i=1}^s$ and weights $\{b_i\}_{i=1}^s$ defined by (3.12). The order of the local error depends on the accuracy of the underlying quadrature formula given by these nodes and weights. The following result is a simple consequence of Theorem 4.3 in [6].[2]

Theorem 3.2. *Let ψ_h be the method defined by* (3.13)-(3.15). *The order of the local error is at least*

$$p = \min(r, 2r-2s+2)$$

where r is the largest integer such that $\sum_{i=1}^{s} b_i c_i^{q-1} = \frac{1}{q}$ for all $1 \le q \le r$. This means that there are positive constants C and h_0 such that

$$d(\psi_h(u), \varphi_h(u)) \le C h^{p+1} \quad \textit{for } h < h_0,\ u \in M.$$

Proof. Choose h small enough such that the solution can be represented in the form $u(h\xi) = \phi_c(\gamma(\xi h)), \xi \in [0,1]$, and consider the corresponding differential equation for γ in T_cM:

$$\frac{d}{dt}\gamma(t) = \left(\phi_c^* F\right)(\gamma(t)) = \left(T_{\gamma(t)}\phi_c\right)^{-1} \Omega\, \mathrm{grad} H\left(\phi_c(\gamma(t))\right). \tag{3.18}$$

Notice that $\left(T_\gamma \phi_c\right)^{-1} = T_U \phi_c^{-1}$ where $U = \phi_c \circ \gamma$ and $T_{U(t)}\phi_c^{-1} : T_{U(t)}M \to T_cM$ for every t. We obtain

$$\frac{d}{dt}\gamma(t) = T_{U(t)}\phi_c^{-1}\Omega \left(T_{U(t)}\phi_c^{-1}\right)^{\mathrm{T}} \left(T_{\gamma(t)}\phi_c\right)^{\mathrm{T}} \mathrm{grad} H\left(\phi_c(\gamma(t))\right). \tag{3.19}$$

Considering the Hamiltonian $\widetilde{H} : T_cM \to \mathbb{R}$, $\widetilde{H}(\gamma) := \phi_c^* H(\gamma) = H \circ \phi_c(\gamma)$, we can then rewrite (3.18) in the form

$$\frac{d}{dt}\gamma(t) = \widetilde{\Omega}(\gamma)\, \mathrm{grad}\widetilde{H}(\gamma), \quad \widetilde{\Omega}(\gamma) := T_{U(t)}\phi_c^{-1}\Omega \left(T_{U(t)}\phi_c^{-1}\right)^{\mathrm{T}}, \tag{3.20}$$

where we have used that $\mathrm{grad}\widetilde{H} = T_{\gamma(t)}\phi_c^{\mathrm{T}} \mathrm{grad} H\left(\phi_c(\gamma(t))\right)$, which is now a gradient on the linear space T_cM with respect to the metric inherited from M, $g|_c$. Locally in a neighborhood of c, (3.13)-(3.15) applied to (3.20) coincides with the methods of Cohen and Hairer, and therefore the order result [6, Thm 4.3] can be applied. Since the Riemannian distance $d(\cdot,\cdot)$ and any norm in local coordinates are equivalent, the result follows. □

3.4.2 Global error

We prove the following result for the global error of DRG methods.

[2] The local error results of this section are valid for general retractions. For the special choice $\phi_c = \exp_c$, an analysis in a purely Riemannian setting could provide sharper geometric insight into the properties of the error.

Theorem 3.3. *Let $u(t)$ be the exact solution to (3.1) where F is a complete vector field on a connected Riemannian manifold (M, g) with flow $u(t) = \varphi_t(u^0)$. Let ψ_h represent a numerical method $u^{k+1} = \psi_h(u^k)$ whose local error can be bounded for some $p \in \mathbb{N}$ as*

$$d(\psi_{\mathfrak{h}}2(u), \varphi_h(u)) \leq Ch^{p+1} \quad \textit{for all } u \in M.$$

Suppose there is a constant L such that

$$\|\nabla F\|_g \leq L,$$

where ∇ is the Levi-Civita connection and $\|\cdot\|_g$ is the operator norm with respect to the metric g. Then the global error is bounded as

$$d\left(u(kh), u^k\right) \leq \frac{C}{L}(\mathrm{e}^{khL} - 1)h^p \quad \textit{for all } k > 0.$$

Proof. Denoting the global error as $e^k := d(u(kh), u^k)$, the triangle inequality yields

$$e^{k+1} \leq d\left(\varphi_h(u(kh))\right), \varphi_h(u^k)) + d\left(\varphi_h(u^k), \psi_h(u^k)\right).$$

The first term is the error at nh propagated over one step, the second term is the local error. For the first term, we find via a Grönwall type inequality of [16],

$$d\left(\varphi_h(u(kh)), \varphi_h(u^k)\right) \leq \mathrm{e}^{hL} d\left(u(kh), u^k\right) = \mathrm{e}^{hL} e^k.$$

Using the local error estimate for the second term, we get the recursion

$$e^{k+1} \leq \mathrm{e}^{hL} e^k + Ch^{p+1},$$

which yields

$$e^k \leq C \frac{\mathrm{e}^{khL} - 1}{\mathrm{e}^{hL} - 1} h^{p+1} \leq \frac{C}{L}(\mathrm{e}^{khL} - 1)h^p.$$

□

Remark: Following Theorem 1.4 in [16], the condition that F is complete can be relaxed if $\varphi_t(u^0)$ and $\{u^k\}_{k\in\mathbb{N}}$ lie in a relatively compact submanifold N of M containing all the geodesics from u^k to $\varphi_{kh}(u^0)$. This is the case if, for instance, H has compact, geodesically convex sublevel sets, since both $\varphi_t(u^0)$ and $\{u^k\}_{k\in\mathbb{N}}$ are restricted to the level set $M_{H(u^0)} = \{p \in M \,|\, H(p) = H(u^0)\}$ and hence lie in the sublevel set $N_{H(u^0)} = \{p \in M \,|\, H(p) \leq H(u^0)\}$.

3.5 Examples and numerical results

To demonstrate how to construct schemes of the type presented, we consider first an example on the two-sphere. The AVF DRG and Midpoint DRG schemes presented in this example could also be obtained by the discrete differential methods on homogeneous manifolds presented in [5]. The novel schemes here are the Itoh–Abe DRG scheme and its symmetrized variant, and the higher order methods obtained by composition or collocation techniques. Then, to demonstrate the usefulness of our methods for problems on more challenging manifolds, we consider first the motion of a particle under gravity on a paraboloid, and then a conservative system on the Stiefel manifold.

3.5.1 Example 1: Perturbed spinning top

We consider a nonlinear perturbation of a spinning top, see [22]. This is a body whose orientation is represented by a vector s of unit length in $\mathbb{R}^3$, so that s lies on the manifold $M = S^2 = \left\{s \in \mathbb{R}^3 : \|s\| = 1\right\}$. Here and in what follows, $\|\cdot\|$ denotes the 2-norm. The ODE system can be written in the form

$$\frac{\mathrm{d}s}{\mathrm{d}t} = \Omega(s)\,\mathrm{grad}H(s), \quad s \in S^2, \quad H \in \mathcal{F}\left(S^2\right), \tag{3.21}$$

where $\Omega(s)y = s \times y$. Given the inertia tensor $\mathbb{I} = \mathrm{diag}(\mathbb{I}_1, \mathbb{I}_2, \mathbb{I}_3)$, and denoting by s^2 the component-wise square of s, we consider the Hamiltonian

$$H(s) = \frac{1}{2}(\mathbb{I}^{-1}s)^{\mathrm{T}}(s + \frac{2}{3}s^2).$$

Geometric integrators for spin systems are discussed widely in the literature, see e.g. [9, 20–22] and references therein. The two-sphere has a simple geometry which makes it attractive for illustrating our new schemes while at the same time being different from Euclidean space, where the standard discrete gradient schemes can be used.

The Riemannian metric g on S^2 restricts to the so-called round metric, coinciding with the Euclidean inner product on the tangent plane of the sphere. Our choice of retraction ϕ is as in [5], given by its restriction to p,

$$\phi_p(x) = \frac{p + x}{\|p + x\|}, \tag{3.22}$$

with the inverse

$$\phi_p^{-1}(u) = \frac{u}{p^{\mathrm{T}}u} - p$$

defined when $p^{\mathrm{T}}u > 0$. We note that $p^{\mathrm{T}}x = 0$ for all $x \in T_pS^2$. The derivative of the retraction and its inverse are given by

$$\begin{aligned} D_x\phi_p &= \frac{1}{\|p+x\|}\left(I - \frac{(p+x)\otimes(p+x)}{\|p+x\|^2}\right), \\ D_u\phi_p^{-1} &= \frac{1}{p^{\mathrm{T}}u}\left(I - \frac{u\otimes p}{p^{\mathrm{T}}u}\right), \end{aligned} \tag{3.23}$$

where ⊗ denotes the outer product[3] of the vectors.

We approximate the system (3.21) numerically, testing the scheme (3.6)-(3.7) with different discrete Riemannian gradients: the AVF (3.8), the midpoint (3.9), the Itoh–Abe (3.10) and its symmetrized version (3.17). For the three symmetric methods, we have chosen $c(s,\widetilde{s}) = \frac{s+\widetilde{s}}{\|s+\widetilde{s}\|}$, so that $\phi_c^{-1}(\widetilde{s}) = -\phi_c^{-1}(s)$. Using that $\mathrm{grad}H(s) = \mathbb{I}^{-1}(s+s^2)$ and considering the transpose of $T_{\gamma_\xi}\phi_c$ from (3.23), the AVF DRG becomes

$$\begin{aligned} \overline{\mathrm{grad}}_{\mathrm{AVF}}H(s,\widetilde{s}) &= \int_0^1 \frac{1}{\|l_\xi\|}\left(I - \frac{l_\xi \otimes l_\xi}{\|l_\xi\|^2}\right)\mathbb{I}^{-1}(\phi_c(\gamma_\xi)+\phi_c(\gamma_\xi)^2)\,\mathrm{d}\xi \\ &= \int_0^1 \frac{1}{\|l_\xi\|}\left(\mathbb{I}^{-1}\left(\phi_c(\gamma_\xi)+\phi_c(\gamma_\xi)^2\right) - \phi_c(\gamma_\xi)^{\mathrm{T}}\mathbb{I}^{-1}\left(\phi_c(\gamma_\xi)+\phi_c(\gamma_\xi)^2\right)\phi_c(\gamma_\xi)\right)\mathrm{d}\xi, \end{aligned}$$

with $\gamma_\xi = (1-\xi)\phi_c^{-1}(s) + \xi\phi_c^{-1}(\widetilde{s}) = (1-2\xi)\phi_c^{-1}(s)$ and $l_\xi = c + \gamma_\xi$. Similarly, the midpoint DRG becomes

$$\overline{\mathrm{grad}}_{\mathrm{MP}}H(s,\widetilde{s}) = \frac{\mathbb{I}^{-1}\left(s+\widetilde{s}+\frac{2}{3}\left(s^2+s\widetilde{s}+\widetilde{s}^2\right)\right) + \frac{\frac{1}{2}\|s+\widetilde{s}\|^2-2}{\|\widetilde{s}-s\|^2}\left(H(\widetilde{s})-H(s)\right)(\widetilde{s}-s)}{\|s+\widetilde{s}\|},$$

where we have used that $g(s,s) = s^{\mathrm{T}}s = 1$ for all $s \in S^2$. To obtain the basis of T_cM for the definition of the Itoh–Abe DRG, we have used the singular-value decomposition. For the first order scheme, noting that $\phi_s^{-1}(s) = 0$, we choose $c(s,\widetilde{s}) = s$, and get $\alpha_j = \phi_s^{-1}(\widetilde{s})^{\mathrm{T}}E_j$, for $j = 1,2$. Then the DRG (3.10) can be written as

$$\overline{\mathrm{grad}}_{\mathrm{IA}}H(s,\widetilde{s}) = \frac{H(s')-H(s)}{\phi_s^{-1}(\widetilde{s})^{\mathrm{T}}E_1}E_1 + \frac{H(\widetilde{s})-H(s)}{\phi_s^{-1}(\widetilde{s})^{\mathrm{T}}E_2}E_2, \tag{3.24}$$

where $s' = \phi_s((\phi_s^{-1}(\widetilde{s})^{\mathrm{T}}E_1)E_1)$.

We solve the same problem using the 4th, 6th and 8th order variants of the collocation-like scheme (3.13)-(3.15). Choosing in the 4th order case the

[3] If x and y are in $\mathbb{R}^3$, $x \otimes y$ is the matrix-matrix product of x taken as a 3×1 matrix and y taken as a 1×3 matrix.

Gaussian nodes $c_{1,2} = \frac{1}{2} \mp \frac{\sqrt{3}}{6}$ as collocation points and setting $c(s, \widetilde{s}) = s$, we get the nonlinear system

$$\begin{aligned}
S_1 &= h\phi_{s_0}\left(\frac{1}{2}\,T_{S_1}\phi_{s_0}^{-1}\left(\Omega_1\,\mathrm{grad}_1 H\right) + \left(\frac{1}{2} - \frac{\sqrt{3}}{3}\right) T_{S_2}\phi_{s_0}^{-1}\left(\Omega_2\,\mathrm{grad}_2 H\right)\right),\\
S_2 &= h\phi_{s_0}\left(\left(\frac{1}{2} + \frac{\sqrt{3}}{3}\right) T_{S_1}\phi_{s_0}^{-1}\left(\Omega_1\,\mathrm{grad}_1 H\right) + \frac{1}{2}\,T_{S_2}\phi_{s_0}^{-1}\left(\Omega_2\,\mathrm{grad}_2 H\right)\right),\\
s_1 &= h\phi_{s_0}\left(T_{S_1}\phi_{s_0}^{-1}\left(\Omega_1\,\mathrm{grad}_1 H\right) + T_{S_2}\phi_{s_0}^{-1}\left(\Omega_2\,\mathrm{grad}_2 H\right)\right),
\end{aligned}$$

where

$$\begin{aligned}
\sigma(\xi h) = &\left(\left(3 + 2\sqrt{3}\right)\phi_{s_0}^{-1}(S_1) + \left(3 - 2\sqrt{3}\right)\phi_{s_0}^{-1}(S_2)\right)\xi\\
&+ \left(3\left(\sqrt{3} - 1\right)\phi_{s_0}^{-1}(S_2) - 3\left(1 + \sqrt{3}\right)\phi_{s_0}^{-1}(S_1)\right)\xi^2,
\end{aligned}$$

and we use the transposes of (3.23) and $\mathrm{grad} H(s) = \mathbb{I}^{-1}(s + s^2)$ in the evaluation of $\mathrm{grad}_1 H$ and $\mathrm{grad}_2 H$. The 6th and 8th order schemes are derived in a similar manner, using the standard Gaussian nodes.

A second order scheme is derived by composing the Itoh–Abe DRG method with its adjoint, and a 4th order scheme is obtained by composing this method again with itself, as well as one by composition of the symmetrized Itoh–Abe DRG method with itself. In all stages of these composition methods, a symmetric $c(u, v)$ is used.

Plots confirming the order of all methods can be seen in Figure 3.1, where solutions using the different schemes are compared to a reference solution obtained using a comparatively small step size. See the left hand panel of Figure 3.2 for numerical confirmation that our methods do indeed preserve the energy to machine precision, while the implicit midpoint method does not. In the right hand panel of Figure 3.2, the solution obtained by the Itoh–Abe DRG scheme with a step size $h = 1$ is plotted together with a solution obtained using the symmetrized Itoh–Abe DRG method with a much smaller time step. We observe, as expected for a method that conserves both the energy and the angular momentum, that the solution stays on the trajectory of the exact solution, although not necessarily at the right place on the trajectory at any given time.

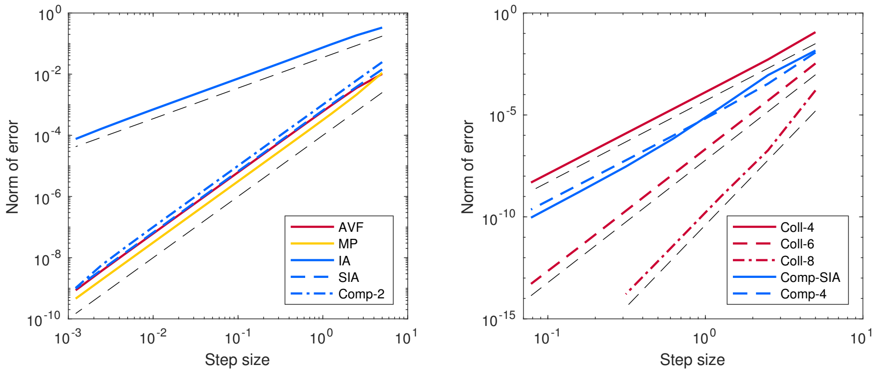


Figure 3.1: Error norm at $t = 10$ for the perturbed spinning top problem solved with different schemes, plotted with black, dashed reference lines of order 1, 2, 4, 6 and 8. Initial condition $s = (-1, -1, 1)/\sqrt{3}$ and $\mathbb{I} = \mathrm{diag}(1, 2, 4)$. *Left:* The AVF, midpoint (MP), Itoh–Abe (IA) and symmetrized Itoh–Abe (SIA) DRGs and a 3-stage composition of the IA DRG scheme (Comp-2). *Right:* Collocation-type schemes of order 4, 6 and 8, a 3-stage composition of the SIA DRG scheme (Comp-SIA), and a 6-stage composition of the IA DRG scheme (Comp-4).

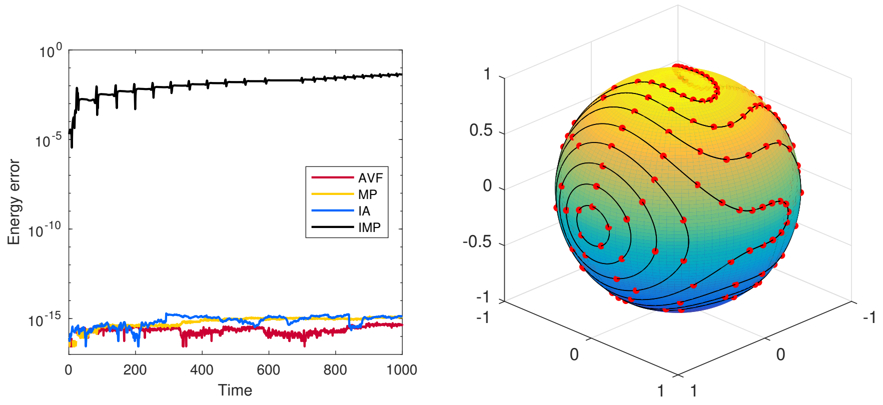


Figure 3.2: *Left:* Energy error with increasing time for the AVF, midpoint (MP) and Itoh–Abe (IA) DRG methods, as well as the implicit midpoint (IMP) method, with step size $h = 1$, initial condition $s = (-1, -1, 1)/\sqrt{3}$ and $\mathbb{I} = \mathrm{diag}(1, 2, 4)$. *Right:* Curves of constant energy on the sphere, found by our method with different starting values. The black solid line is the solution using the symmetrized Itoh–Abe DRG method with step size $h = 0.01$, while the red dots are the solutions obtained by the Itoh–Abe DRG method with step size $h = 1$.

3.5.2 Example 2: Particle moving under gravity on a paraboloid

We consider a particle of unit mass moving under a gravitational field on an elliptic paraboloid given by

$$P = \left\{ q \in \mathbb{R}^3 : \frac{x(q)^2}{a^2} + \frac{y(q)^2}{b^2} - 2z(q) = d \right\},$$

where x, y, z are the Cartesian coordinate functions, and $a, b, d \in \mathbb{R}^+$. See [29, pp. 106-108] and [11] on discussion of such a system and the existence of solutions. Using, as in [18], an inertial Euclidean frame to express the position of the particle in $\mathbb{R}^3$, we obtain the Hamiltonian

$$H(q,p) = \frac{1}{2}p^T p + g q_3,$$

where $p = \frac{\partial L}{\partial \dot{q}} = \dot{q}$ are the momentum coordinates and g is the gravitational constant. Thus the dynamics can be described on the cotangent bundle $T^*P =: M$ by the Hamiltonian equations, for $i = 1,2,3$,

$$\dot{q}_i = \frac{\partial H}{\partial p_i}(q,p), \qquad \dot{p}_i = -\frac{\partial H}{\partial q_i}(q,p), \qquad q \in P, \quad p \in T_q P.$$

We define the retraction by its restriction to a center point $c = (c_P, c_T)$, $c_P \in P$, $c_T \in T_{c_P}P$, so that $\phi : TM \to M$ is given by $\phi_c(u,v) = (\phi_{P,c}(u), \phi_{T,c,u}(v))$, where $\phi_{P,c} : T_{c_P}P \to P$ and $\phi_{T,c,u} : T_{c_T}T_{c_P}P \to T_{\phi_{P,c}(u)}P$. Our choice of $\phi_{P,c}(u)$ is the projection onto the paraboloid P along the straight line in $\mathbb{R}^3$ from $c_P + u$ to the origin. The second component $\phi_{T,c}(u,v)$ is the linear projection in $\mathbb{R}^3$ of $c_T + v$ to $T_{c_P}P$. That is,

$$\begin{aligned}
\phi_{P,c}(u) &= \frac{d}{\alpha - c_{P,3} - u_3}(c_P + u),\\
\alpha &= \sqrt{(c_{P,3} + u_3)^2 + d\left(\frac{(c_{P,1} + u_1)^2}{a^2} + \frac{(c_{P,2} + u_2)^2}{b^2}\right)},\\
\phi_{T,c,u}(v) &= c_T + v - \frac{\beta(\phi_{P,c}(u))^T(c_T + v)}{\beta(\phi_{P,c}(u))^T\beta(\phi_{P,c}(u))}\beta(\phi_{P,c}(u)),\\
\beta(q) &= \left(\frac{q_1}{a^2}, \frac{q_2}{b^2}, -1\right)^T.
\end{aligned}$$

This has the inverse $\phi_c^{-1}(q,p) = (\phi_{P,c}^{-1}(q), \phi_{T,c,q}^{-1}(p))$, where

$$\phi_{P,c}^{-1}(q) = \frac{c_{P,3} + d}{\beta(c_P)^T q}q - c_P, \qquad \phi_{T,c,q}^{-1}(p) = p - c_T - \frac{\beta(c_P)^T(p - c_T)}{\beta(c_P)^T\beta(q)}\beta(q).$$

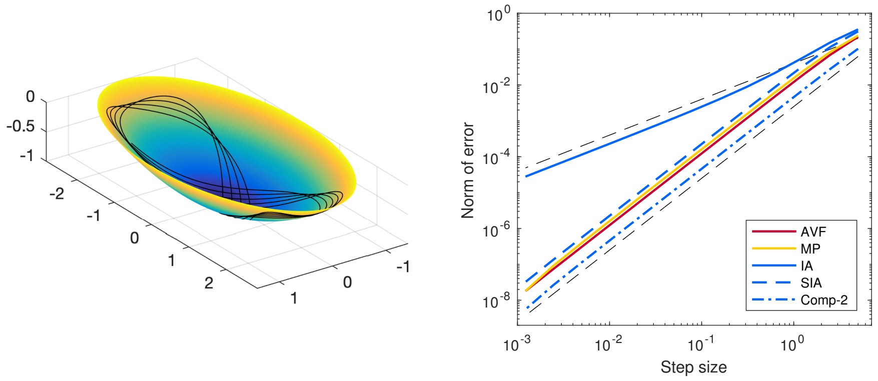


Figure 3.3: Particle moving under gravity on the paraboloid. *Left:* The system solved by the symmetrized Itoh–Abe DRG method with step size $h = 0.01$, from $t_0 = 0$ to $T = 20$. *Right:* Error norm at $t = 1$ for the problem solved with different schemes: The AVF, midpoint (MP), Itoh–Abe (IA) and symmetrized Itoh–Abe (SIA) DRG methods and a 3-stage composition of the IA DRG scheme (Comp-2), plotted against black, dashed reference lines of order 1 and 2.

We test our schemes on the problem with the paraboloid given by $a = 1$, $b = 2$, $d = 2$, and starting values $q = (1/2, 1/2, -27/32)$, $p = (3/2, 7/2, 19/16)$. We compare to the solution of standard methods in $\mathbb{R}^6$ with a comparatively small time step size to confirm numerically that the methods converge to the correct solution. The numerical results show that our schemes have the expected order, see Figure 3.3. Preservation of the Hamiltonian was also observed.

3.5.3 Example 3: Conservative system on the Stiefel manifold

Lastly we consider an ODE

$$\dot{Y} = S(Y)\,\mathrm{grad}H(Y), \qquad Y \in \mathbb{V}_p(\mathbb{R}^n), \qquad H \in \mathcal{F}\Big(\mathbb{V}_p(\mathbb{R}^n)\Big), \tag{3.25}$$

on the Stiefel manifold $M = \mathbb{V}_p(\mathbb{R}^n) = \Big\{Y \in \mathbb{R}^{n\times p} : Y^T Y = I_p\Big\}$, i.e. the set of all $n \times p$ matrices whose p columns are orthonormal. The solution of (3.25) stays on the Stiefel manifold at all times if

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(Y^T Y) = Y^T\dot{Y} + \dot{Y}^T Y = Y^T S(Y)\,\mathrm{grad}H(Y) + \mathrm{grad}H(Y)^T S(Y)^T Y \\ = 0,\end{aligned} \tag{3.26}$$

i.e. if $Y^T S(Y)\,\mathrm{grad}H(Y)$ is a skew-symmetric $p \times p$ matrix. We shall consider a problem on $\mathbb{V}_p(\mathbb{R}^n)$ with first integral

$$H(Y(t)) = H(Y(t_0)), \quad H(Y) = \mathrm{tr}(Y^T Y^2), \tag{3.27}$$

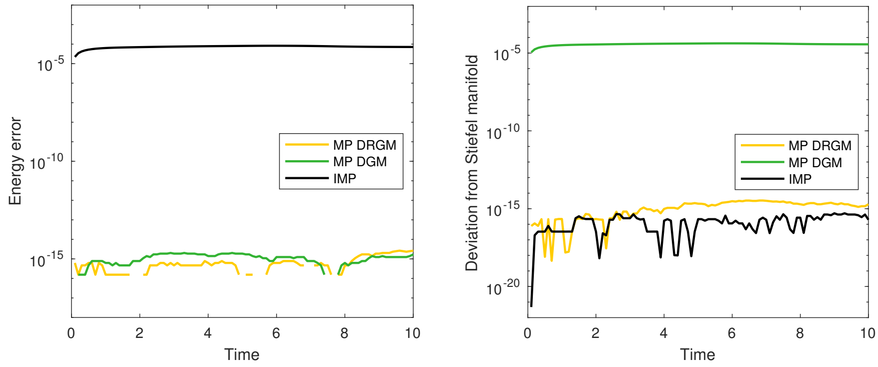


Figure 3.4: Relative energy error (left) and deviation from $\mathbb{V}_p(\mathbb{R}^n)$ (right) with increasing time for the midpoint discrete Riemannian gradient method (MP DRGM), as well as the standard midpoint discrete gradient method (MP DGM) and the implicit midpoint (IMP) method, with step size $h = 0.1$, $n = 5$ and $p = 2$. The relative energy error in step k is measured by $|(H(Y_k) - H(Y_0))/H(Y_0)|$, while the deviation from the Stiefel manifold is measured by $\|I_p - Y_k^T Y_k\|_F$, where $\|\cdot\|_F$ denotes the Frobenius norm.

where Y^2 means the component-wise square of Y. In (3.25), H is a first integral whenever $S(Y) \in \mathbb{R}^{n\times n}$ fulfills (3.26) as well as being skew-symmetric with respect to the Riemannian metric g,

$$g_Y(U,V) = \mathrm{tr}(U^T(I_n - \frac{1}{2}YY^T)V), \quad U,V \in T_Y M,$$

named the canonical metric by Edelman et al. in [8]. The Riemannian gradient of H follows from this; it is the tangent vector grad$H$ satisfying $g_Y(\mathrm{grad}H, V) = \mathrm{tr}(\nabla H(Y)^T V)$ for all $V \in T_Y M$, where $\nabla H(Y)$ denotes the Euclidean gradient. That is, as stated in [8],

$$\mathrm{grad}H(Y) = \nabla H(Y) - Y\nabla H(Y)^T Y.$$

The intrinsic Riemannian structures provide the components needed in a DRG scheme. We choose the Riemannian exponential as retraction. That is, $\phi_C(V)$ is given by going the distance 1 along the geodesic path emanating from the base point $C \in M$ in the direction $V \in T_C M$. Similarly, the inverse retraction is given by the Riemannian logarithm, and the base point $C(Y,\tilde{Y})$ is given by the geodesic midpoint between Y and $\tilde{Y}$. To calculate the geodesic and the Riemannian exponential, we use the method introduced by Edelman et al. in [8, Corollary 2.2]. For the logarithm, we use the algorithm of Zimmermann [30].

Any numerical method preserving quadratic invariants, like symplectic Runge–Kutta methods, will find solutions on $\mathbb{V}_p(\mathbb{R}^n)$. However, such a method will in general not preserve the cubic invariant (3.27). A standard discrete gradient method can be implemented to preserve either (3.26) or (3.27), but not both. For our numerical experiments, we have considered (3.25) with $n = 5$ and $p = 2$, and $S(Y)$ chosen so that $H(Y)$ is conserved. As demonstrated in Figure 3.4, a DRG method can be used to get solutions that stay on the Stiefel manifold while preserving the first integral.

3.6 Conclusions and further work

We have presented a general framework for constructing energy preserving numerical integrators on Riemannian manifolds. The main tool is to generalize the notion of discrete gradients as known from the literature. The new methods make use of an approximation to the Riemannian gradient coined the discrete Riemannian gradient, as well as a retraction map and a coordinate center function.

Particular examples of discrete Riemannian gradient methods are given as generalizations of well-known schemes, such as the average vector field method, the midpoint discrete gradient method and the Itoh–Abe method. Extensions to higher order are proposed via a collocation-like method. We have analysed the local and global error behaviour of the methods, and they have been implemented and tested for problems on the two-sphere, the paraboloid and the Stiefel manifold.

Possible directions for future research include a more detailed study of the stability and propagation of errors, taking into account particular features of the Riemannian manifold; for instance, it may be expected that the sectional curvature will play an important role. We believe, inspired by [3], that there is a potential for making our implementations more efficient by tailoring them to the particular manifold, as well as the ODE problem considered.

Bibliography

[1] R. L. Adler, J.-P. Dedieu, J. Y. Margulies, M. Martens, and M. Shub. Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA J. Numer. Anal.*, 22(3):359–390, 2002.

[2] L. Brugnano, F. Iavernaro, and D. Trigiante. Hamiltonian boundary value methods (energy preserving discrete line integral methods). *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, 5(1):17–37, 2010.

[3] E. Celledoni, S. Eidnes, B. Owren, and T. Ringholm. Dissipative numerical schemes on Riemannian manifolds with applications to gradient flows. *SIAM J. Sci. Comput.*, 40(6):A3789–A3806, 2018.

[4] E. Celledoni, H. Marthinsen, and B. Owren. An introduction to Lie group integrators – basics, new developments and applications. *J. Comput. Phys.*, 257:1040–1061, 2014.

[5] E. Celledoni and B. Owren. Preserving first integrals with symmetric Lie group methods. *Discrete Contin. Dyn. Syst.*, 34(3):977–990, 2014.

[6] D. Cohen and E. Hairer. Linear energy-preserving integrators for Poisson systems. *BIT*, 51(1):91–101, 2011.

[7] P. E. Crouch and R. Grossman. Numerical integration of ordinary differential equations on manifolds. *J. Nonlinear Sci.*, 3(1):1–33, 1993.

[8] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1999.

[9] J. Frank, W. Huang, and B. Leimkuhler. Geometric integrators for classical spin systems. *J. Comput. Phys.*, 133(1):160–172, 1997.

[10] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[11] A. Gray, A. Jones, and R. Rimmer. Motion under gravity on a paraboloid. *J. Differential Equations*, 45(2):168–181, 1982.

[12] E. Hairer. Energy-preserving variant of collocation methods. *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, 5(1-2):73–84, 2010.

[13] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, 25(1):35–61, 1983.

[14] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numer.*, 9:215–365, 2000.

[15] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 76(1):85–102, 1988.

[16] M. Kunzinger, H. Schichl, R. Steinbauer, and J. A. Vickers. Global Gronwall estimates for integral curves on Riemannian manifolds. *Rev. Mat. Complut.*, 19(1):133–137, 2006.

[17] J. M. Lee. *Riemannian manifolds*, volume 176 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1997.

[18] T. Lee, M. Leok, and N. H. McClamroch. *Global formulations of Lagrangian and Hamiltonian dynamics on manifolds*. Interaction of Mechanics and Mathematics. Springer, Cham, 2018.

[19] B. Leimkuhler and G. W. Patrick. A symplectic integrator for Riemannian manifolds. *J. Nonlinear Sci.*, 6(4):367–384, 1996.

[20] D. Lewis and N. Nigam. Geometric integration on spheres and some interesting applications. *J. Comput. Appl. Math.*, 151(1):141–170, 2003.

[21] R. McLachlan, K. Modin, and O. Verdier. A minimal-variable symplectic integrator on spheres. *Math. Comp.*, 86(307):2325–2344, 2017.

[22] R. I. McLachlan, K. Modin, and O. Verdier. Symplectic integrators for spin systems. *Phys. Rev. E*, 89(6):061301, 2014.

[23] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *Philos. Trans. Roy. Soc. A*, 357(1754):1021–1045, 1999.

[24] J. Milnor. *Morse theory*. Based on lecture notes by M. Spivak and R. Wells. Annals of Mathematics Studies, No. 51. Princeton University Press, Princeton, N.J., 1963.

[25] H. Munthe-Kaas. Lie–Butcher theory for Runge–Kutta methods. *BIT*, 35(4):572–587, 1995.

[26] H. Munthe-Kaas. Runge–Kutta methods on Lie groups. *BIT*, 38(1):92–111, 1998.

[27] H. Z. Munthe-Kaas and A. Lundervold. On post-Lie algebras, Lie–Butcher series and moving frames. *Found. Comput. Math.*, 13(4):583–613, 2013.

[28] G. Quispel and D. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A*, 41(4):045206, 7, 2008.

[29] E. T. Whittaker. *A treatise on the analytical dynamics of particles and rigid bodies*. Cambridge Mathematical Library. Cambridge University Press, Cambridge, 1988. Reprint of the 1937 edition.

[30] R. Zimmermann. A matrix-algebraic algorithm for the Riemannian logarithm on the Stiefel manifold under the canonical metric. *SIAM J. Matrix Anal. Appl.*, 38(2):322–342, 2017.

Dissipative numerical schemes on Riemannian manifolds with applications to gradient flows

Elena Celledoni, Sølve Eidnes, Brynjulf Owren and Torbjørn Ringholm

Published in SIAM Journal on Scientific Computing 40 (6), pp. A3789–A3806 (2018)

Dissipative numerical schemes on Riemannian manifolds with applications to gradient flows

Abstract. This paper concerns an extension of discrete gradient methods to finite-dimensional Riemannian manifolds termed discrete Riemannian gradients, and their application to dissipative ordinary differential equations. This includes Riemannian gradient flow systems which occur naturally in optimization problems. The Itoh–Abe discrete gradient is formulated and applied to gradient systems, yielding a derivative-free optimization algorithm. The algorithm is tested on two eigenvalue problems and two problems from manifold valued imaging: InSAR denoising and DTI denoising.

4.1 Introduction

When designing and applying numerical schemes for solving systems of ODEs and PDEs there are several important properties which serve to distinguish schemes, one of which is the preservation of geometric features of the original system. The field of geometric integration encompasses many types of numerical schemes for ODEs and PDEs specifically designed to preserve one or more such geometric features; a non-exhaustive list of features includes symmetry, symplecticity, first integrals (or energy), orthogonality, and manifold structures such as Lie group structure [14]. Energy conserving methods have a successful history in the field of numerical integration of ODEs and PDEs. In a similar vein, numerical schemes with guaranteed dissipation are useful for solving dissipative equations such as gradient systems.

As seen in [17], any Runge–Kutta method can be dissipative when applied to gradient systems as long as step sizes are chosen small enough; less severe but still restrictive conditions for dissipation in Runge–Kutta methods are presented in [13]. In [10], Gonzalez introduces the notion of discrete gradient schemes with energy preserving properties, later expanded upon to include dissipative systems in [21]. These articles consider ODEs in Euclidian spaces only with the exception of [13] where the authors also consider Runge–Kutta methods on manifolds defined by constraints. Unlike the Runge–Kutta methods, discrete gradient methods are dissipative for all step sizes, meaning one can employ adaptive time steps while retaining convergence toward fixed points [25]. However, one may experience a practical step size restriction when applying discrete gradient methods to very stiff problems, due to the lack of L-stability as seen when applying the Gonzalez and mean value discrete gradients to problems with quadratic potentials [13] [15]. Motivated by their work on Lie group methods, the energy conserving discrete gradient method was generalized to ODEs on manifolds, and Lie groups particularly, in [7] where the authors introduce the concept of discrete differentials. In [5], this concept is specialized in the setting of Riemannian manifolds. To the best of our knowledge, the discrete gradient methods have not yet been formulated for dissipative ODEs on manifolds. Doing so is the central purpose of this article.

One of the main reasons for generalizing discrete gradient methods to dissipative systems on manifolds is that gradient systems are dissipative, and gradient flows are natural tools for optimization problems which arise in e.g. manifold-valued image processing and eigenvalue problems. The goal is then to find one or more stationary points of the gradient flow of a functional $V : M \to \mathbb{R}$, which correspond to critical points of V. This approach is, among other optimization methods, presented in [1]. Since gradient systems occur naturally on Riemannian manifolds, it is natural to develop our schemes in a Riemannian manifold setting.

A similarity between the optimization algorithms in [1] and the manifold valued discrete gradient methods in [7] is their use of retraction mappings. Retraction mappings were introduced for numerical methods in [26], see also [2]; they are intended as computationally efficient alternatives to parallel transport on manifolds. Our methods will be formulated as a framework using general discrete gradients on general Riemannian manifolds with general retractions. We will consider a number of specific examples that illustrate how to apply the procedure in practical problems.

As detailed in [11] and [22], using the Itoh–Abe discrete gradient [18], one can obtain an optimization scheme for n-dimensional problems with a limited degree of implicitness. At every iteration, one needs to solve n decoupled scalar nonlinear subequations, amounting to $\mathcal{O}(n)$ operations per step. In other discrete gradient schemes a system of n coupled nonlinear equations must be solved per iteration, amounting to $\mathcal{O}(n^2)$ operations per step. The Itoh–Abe discrete gradient method therefore appears to be well suited to large-scale problems such as image analysis problems, and so it seems natural to apply our new methods to image analysis problems on manifolds, see Section 4.4.2. In [7], the authors generalize the average vector field [16] and midpoint [10] discrete gradients, but not the Itoh–Abe discrete gradient, to Lie groups and homogeneous manifolds. A novelty of this article is the formulation of the Itoh–Abe discrete gradient for problems on manifolds.

As examples we will consider two eigenvalue finding problems, in addition to the more involved problems of denoising InSAR and DTI images using total variation (TV) regularization [30]. The latter two problems we consider as real applications of the algorithm. The two eigenvalue problems are included mostly for the exposition and illustration of our methods, as well as for testing convergence properties.

The paper is organized as follows: Below, we introduce notation and fix some fundamental definitions used later on. In the next section, we formulate the dissipative problems we wish to solve. In section 3, we present the discrete Riemannian gradient (DRG) methods, a convergence proof for the family of optimization methods obtained by applying DRG methods to Riemannian gradient flow problems, the Itoh–Abe discrete gradient generalized to manifolds, and the optimization algorithm obtained by applying the Itoh–Abe DRG to the gradient flow problem. In section 4, we provide numerical experiments to illustrate the use of DRGs in optimization, and in the final section we present conclusions and avenues for future work.

Notation and preliminaries

Some notation and definitions used in the following are summarized below. For a more thorough introduction to the concepts, see e.g. [19] or [20].

Table 4.1: Notational conventions

Notation	Description
M	n-dimensional Riemannian manifold
T_pM	tangent space at $p \in M$ with zero vector 0_p
T_p^*M	cotangent space at $p \in M$
TM	tangent bundle of M
T^*M	cotangent bundle of M
$\mathfrak{X}(M)$	space of vector fields on M
$g(\cdot,\cdot)$	Riemannian metric on M
$\Vert\cdot\Vert_p$	Norm induced on T_pM by g
$\{E_l\}_{l=1}^n$	g-orthogonal basis of T_pM

On any differentiable manifold there is a duality pairing $\langle\cdot,\cdot\rangle : T^*M \times TM \to \mathbb{R}$ which we will denote as $\langle\omega, v\rangle = \omega(v)$. Furthermore, the Riemannian metric sets up an isomorphism between TM and T^*M via the linear map $v \mapsto g(v,\cdot)$. This map and its inverse, termed the musical isomorphisms, are known as the flat map $^\flat : TM \to T^*M$ and sharp map $^\sharp : T^*M \to TM$, respectively. The applications of these maps are also termed index raising and lowering when considering the tensorial representation of the Riemannian metric. Note that with the above notation we have the idiom $x^\flat(y) = \left\langle x^\flat, y\right\rangle = g(x, y)$.

On a Riemannian manifold, one can define gradients: For $V \in C^\infty(M)$, the (Riemannian) gradient with respect to g, $\operatorname{grad}_g V \in \mathfrak{X}(M)$, is the unique vector field such that $g(\operatorname{grad}_g V, X) = \left\langle \mathrm{d}V, X\right\rangle$ for all $X \in \mathfrak{X}(M)$. In the language of musical isomorphisms, $\operatorname{grad}_g V = (\mathrm{d}V)^\sharp$. For the remainder of this article, we will write $\mathrm{grad} V$ for the gradient and assume that it is clear from the context which g is to be used.

Furthermore, the *geodesic* between p and q is the unique curve of minimal length between p and q, providing a distance function $d_M : M \times M \to \mathbb{R}$. The geodesic γ passing through p with tangent v is given by the Riemannian exponential at p, $\gamma(t) = \exp_p(tv)$. For any p, $\exp_p$ is a diffeomorphism on a neighbourhood N_p of 0_p, The image $\exp_p(S_p)$ of any star-shaped subset $S_p \subset N_p$ is called a normal neighbourhood of p, and on this, $\exp_p$ is a radial isometry, i.e. $d_M(p, \exp_p(v)) = \|v\|_p$ for all $v \in S_p$.

4.2 The problem

We will consider ordinary differential equations (ODEs) of the form

$$\dot{u} = F(u), \quad u(0) = u^0 \in M, \tag{4.1}$$

where $F \in \mathfrak{X}(M)$ has an associated energy $V : M \to \mathbb{R}$ dissipating along solutions of (4.1). That is, with $u(t)$ a solution of (4.1):

$$\frac{\mathrm{d}}{\mathrm{d}t} V(u) = \left\langle \mathrm{d}V(u), \dot{u} \right\rangle = \left\langle \mathrm{d}V(u), F(u) \right\rangle = g(\mathrm{grad} V(u), F(u)) \leq 0.$$

An example of such an ODE is the gradient flow. Given an energy V, the gradient flow of V with respect to a Riemannian metric g is

$$\dot{u} = -\mathrm{grad} V(u), \tag{4.2}$$

which is dissipative since if $u(t)$ solves (4.2), we have

$$\frac{\mathrm{d}}{\mathrm{d}t} V(u) = -g\left(\mathrm{grad} V(u), \mathrm{grad} V(u)\right) \leq 0.$$

Remark: This setting can be generalized by an approach similar to [21]. Suppose there exists a (0,2) tensor field h on M such that $h(x,x) \leq 0$. We can associate to h the (1,1) tensor field $H : TM \to TM$ given by $Hx = h(x,\cdot)^\sharp$. Consider the system

$$\dot{u} = H \mathrm{grad} V(u). \tag{4.3}$$

This system dissipates V, since

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} V(u) &= \left\langle \mathrm{d}V(u), H \mathrm{grad} V(u) \right\rangle \\ &= g\left(\mathrm{grad} V(u), H \mathrm{grad} V(u)\right) \\ &= h\left(\mathrm{grad} V(u), \mathrm{grad} V(u)\right) \leq 0. \end{aligned}$$

Any dissipative system of the form (4.1) can be written in this form on the set $M \backslash \{p \in M : g(F(p), \mathrm{grad} V(p)) = 0\}$ since, given F and V, we can construct h as follows:

$$h = \frac{1}{g(F, \mathrm{grad} V)} F^\flat \otimes F^\flat.$$

If $F = -\mathrm{grad} V$, we take $h = -g$ such that H becomes $-\mathrm{Id}$, and recover (4.2). In the following, we mainly discuss the case $F = -\mathrm{grad} V$ for the sake of notational clarity.

4.3 Numerical scheme

The discrete differentials in [7] are formulated such that they may be used on non-Riemannian manifolds. Since we restrict ourselves to Riemannian manifolds, we define their analogues: discrete Riemannian gradients. As with the discrete differentials, we shall make use of retractions as defined in [26].

Definition 4.1. Let $\phi : TM \to M$ and denote by ϕ_p the restriction of ϕ to T_pM. Then, ϕ is a *retraction* if the following conditions are satisfied:

- ϕ_p is smooth and defined in an open ball $B_{r_p(0_p)}$ of radius r_p around 0_p, the zero vector in T_pM.

- $\phi_p(v) = p$ if and only if $v = 0_p$.

- Identifying $T_{0_p} T_pM \simeq T_pM$, ϕ_p satisfies

$$d\phi_p\big|_{0_p} = \mathrm{id}_{T_pM},$$

where id_{T_pM} denotes the identity mapping on T_pM.

From the inverse function theorem it follows that for any p, there exists a neighbourhood $U_{p,\phi} \in T_pM$ of 0_p, such that $\phi_p : U_{p,\phi} \to \phi_p(U_{p,\phi})$ is a diffeomorphism. In general, ϕ_p is not a diffeomorphism on the entirety of T_pM and so all the following schemes must be considered local in nature. The canonical retraction on a Riemannian manifold is the Riemannian exponential. It may be computationally expensive to evaluate even if closed expressions for geodesics are known, and so one often wishes to come up with less costly retractions if possible. We are now ready to introduce the notion of discrete Riemannian gradients.

Definition 4.2. Given a retraction ϕ, a function $c : M \times M \to M$ where $c(p, p) = p$ for all $p \in M$ and a continuous $V : M \to \mathbb{R}$, then $\overline{\mathrm{grad}} V : M \times M \to TM$ is a discrete Riemannian gradient of V if it is continuous and, for all $p, q \in U_{c(p,q),\phi}$,

$$V(q) - V(p) = g\left(\overline{\mathrm{grad}}V(p,q), \phi^{-1}_{c(p,q)}(q) - \phi^{-1}_{c(p,q)}(p)\right) \tag{4.4}$$

$$\overline{\mathrm{grad}}V(p,p) = \mathrm{grad}V|_p. \tag{4.5}$$

We formulate a numerical scheme for equation (4.2) based on this definition. Given times $0 = t_0 < t_1 < ...$, let u^k denote the approximation to $u(t_k)$ and let $\tau_k = t_{k+1} - t_k$. Then, we take

$$u^{k+1} = \phi_{c^k}\left(W(u^k, u^{k+1})\right) \tag{4.6}$$

$$W(u^k, u^{k+1}) = \phi^{-1}_{c^k}(u^k) - \tau_k \overline{\mathrm{grad}}V(u^k, u^{k+1}) \tag{4.7}$$

where $c^k = c(u^k, u^{k+1})$ and In the above and all of the following, we assume that u^k and u^{k+1} lie in $U_{c^k,\phi} \cap S_{c^k}$. The following proposition verifies that the scheme is dissipative.

Proposition 4.1. *The sequence* $\{u^k\}_{k\in\mathbb{N}}$ *generated by the DRG scheme* (4.6)-(4.7) *satisfies* $V(u^{k+1}) - V(u^k) \leq 0$ *for all* $k \in \mathbb{N}$.

Proof. Using property (4.4) and equations (4.6) and (4.7), we get

$$\begin{aligned} V(u^{k+1}) - V(u^k) &= g\left(\overline{\mathrm{grad}}V(u^k, u^{k+1}), \phi^{-1}_{c^k}(u^{k+1}) - \phi^{-1}_{c^k}(u^k)\right) \\ &= g\left(\overline{\mathrm{grad}}V(u^k, u^{k+1}), W(u^k, u^{k+1}) - \phi^{-1}_{c^k}(u^k)\right) \\ &= -\tau_k g\left(\overline{\mathrm{grad}}V(u^k, u^{k+1}), \overline{\mathrm{grad}}V(u^k, u^{k+1})\right) \leq 0 \end{aligned}$$

□

Remark: This extends naturally to schemes for (4.3) by exchanging (4.7) for

$$W(u^k, u^{k+1}) = \phi^{-1}_{c^k}(u^k) + \tau_k \overline{H}_{(u^k,u^{k+1})} \overline{\mathrm{grad}}V(u^k, u^{k+1}),$$

where $\overline{H}_{(p,q)}$ is the (1,1) tensor associated with a negative semi-definite (0,2) tensor field $\overline{h}_{(p,q)} : T_{c(p,q)}M \times T_{c(p,q)}M \to \mathbb{R}$ approximating $h|_p$ consistently.

Two DRGs, the AVF DRG and the Gonzalez DRG, can be easily found by index raising the discrete differentials defined in [7]. We will later generalize the Itoh–Abe discrete gradient, but first we present a proof that the DRG scheme converges to a stationary point when used as an optimization algorithm. We will need the following definition of coercivity:

Definition 4.3. A function $V : M \to \mathbb{R}$ is *coercive* if, for all $v \in M$, every sequence $\{u^k\}_{k\in\mathbb{N}} \subset M$ such that $\lim\limits_{k\to\infty} d_M(u^k, v) = \infty$ also satisfies $\lim\limits_{k\to\infty} V(u^k) = \infty$.

We will also need the following theorem from [28], concerning the boundedness of the sublevel sets $M_\mu = \{u \in M : V(u) \leq \mu\}$ of V:

Theorem 4.1. *Assume M is unbounded. Then the sublevel sets of $V : M \to \mathbb{R}$ are bounded if and only if V is coercive.*

Proof. See [28], Theorem 8.6, Chapter 1 and the remarks below it. □

Equipped with this, we present the following theorem, the proof of which is inspired by that of the convergence theorem in [11].

Theorem 4.2. *Assume that M is geodesically complete, that $V : M \to R$ is coercive, bounded from below and continuously differentiable, and that $\overline{\mathrm{grad}}V$ is continuous. Then, the iterates $\{u^k\}_{k\in\mathbb{N}}$ produced by applying the discrete Riemannian gradient scheme* (4.6)-(4.7) *with time steps $0 < \tau_{min} \leq \tau_k \leq \tau_{max}$ and $c^k = u^k$ or $c^k = u^{k+1}$, to the gradient flow of V satisfy*

$$\lim_{k\to\infty} \overline{\mathrm{grad}}V(u^k, u^{k+1}) = \lim_{k\to\infty} \mathrm{grad}V(u^k) = 0.$$

Additionally, there exists at least one accumulation point u^* of $\{u^k\}_{k\in\mathbb{N}}$, and any such accumulation point satisfies $\mathrm{grad}V(u^*) = 0$.

Proof. Since V is bounded from below and by Proposition 4.1, we have

$$C \leq V(u^{k+1}) \leq V(u^k) \leq \ldots \leq V(u^0)$$

such that, by the monotone convergence theorem, $V^* := \lim_{k\to\infty} V(u^k)$ exists. Furthermore, by property (4.4) and using the scheme (4.6)-(4.7):

$$\begin{aligned}\frac{1}{\tau_k}\left\|\phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1})\right\|_{c^k}^2 &= \tau_k \left\|\overline{\mathrm{grad}}V(u^k, u^{k+1})\right\|_{c^k}^2 \\ &= g\left(\overline{\mathrm{grad}}V(u^k, u^{k+1}), \phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1})\right) \\ &= V(u^k) - V(u^{k+1}).\end{aligned}$$

From this, it is clear that for any $i, j \in \mathbb{N}$,

$$\sum_{k=i}^{j-1} \tau_k \left\|\overline{\mathrm{grad}}V(u^k, u^{k+1})\right\|_{c^k}^2 = V(u^i) - V(u^j) \leq V(u^0) - V^*$$

and

$$\sum_{k=i}^{j-1} \frac{1}{\tau_k}\left\|\phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1})\right\|_{c^k}^2 = V(u^i) - V(u^j) \leq V(u^0) - V^*.$$

In particular,

$$\sum_{k=0}^{\infty} \left\| \overline{\text{grad}} V(u^k, u^{k+1}) \right\|_{c^k}^2 \leq \frac{V(u^0) - V^*}{\tau_{min}},$$

and

$$\sum_{k=0}^{\infty} \left\| \phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1}) \right\|_{c^k}^2 \leq \tau_{max} \Big(V(u^0) - V^* \Big),$$

meaning

$$\lim_{k\to\infty} \left\| \overline{\text{grad}} V(u^k, u^{k+1}) \right\|_{c^k} = 0,$$
$$\lim_{k\to\infty} \left\| \phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1}) \right\|_{c^k} = 0.$$

Since u^{k+1} is in a normal neighbourhood of c^k,

$$d_M(c^k, u^{k+1}) = d_M(c^k, \exp_{c^k}(\exp_{c^k}^{-1}(u^{k+1}))) = \| \exp_{c^k}^{-1}(u^{k+1}) \|_{c^k}. \tag{4.8}$$

Introduce $\psi_{c^k} : T_{c^k}M \to T_{c^k}M$ by $\psi_{c^k} = \exp_{c^k}^{-1} \circ \phi_{c^k}$. Since both exp and ϕ are retractions,

$$\psi_{c^k}(0_{c^k}) = 0_{c^k},$$
$$D\psi_{c^k}|_{0_{c^k}} = \text{id}_{T_{c^k}M}.$$

Thus, per definition of Fréchet derivatives,

$$\psi_{c^k}(x) - \psi_{c^k}(0_{c^k}) - D\psi_{c^k}|_{0_{c^k}} x = \psi_{c^k}(x) - x = o(x),$$

in particular: choosing $x = \phi_{c^k}^{-1}(u^{k+1})$ we get

$$\exp_{c^k}^{-1}(u^{k+1}) - \phi_{c^k}^{-1}(u^{k+1}) = o(\|\phi_{c^k}^{-1}(u^{k+1})\|_{c^k}),$$

meaning

$$\| \exp_{c^k}^{-1}(u^{k+1}) \|_{c^k} \leq \|\phi_{c^k}^{-1}(u^{k+1})\|_{c^k} + o(\|\phi_{c^k}^{-1}(u^{k+1})\|_{c^k}). \tag{4.9}$$

Taking $c^k = u^k$ and combining (4.8) and (4.9) we find

$$d(u^k, u^{k+1}) = \| \exp_{c^k}^{-1}(u^{k+1}) \|_{c^k} \leq \|\phi_{c^k}^{-1}(u^{k+1})\|_{c^k} + o(\|\phi_{c^k}^{-1}(u^{k+1})\|_{c^k}).$$

Hence, since $\left\| \phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1}) \right\|_{c^k} = \left\| \phi_{c^k}^{-1}(u^{k+1}) \right\|_{c^k}$ when $c^k = u^k$,

$$\lim_{k\to\infty} d(u^k, u^{k+1}) \leq \lim_{k\to\infty} \left\| \phi_{c^k}^{-1}(u^k) - \phi_{c^k}^{-1}(u^{k+1}) \right\|_{c^k} = 0. \tag{4.10}$$

Note that we can exchange the roles of u^k and u^{k+1} and obtain the same result.

Since V is bounded from below, the sublevel sets M_μ of V are the preimages of the closed subsets $[C,\mu]$ and are hence closed as well. Since V is assumed to be coercive, by Theorem 4.1 the M_μ are bounded, and so since M is geodesically complete, by the Hopf-Rinow theorem the M_μ are compact [28]. In particular, $M_{V(u^0)}$ is compact such that $\overline{\mathrm{grad}}V$ is uniformly continuous on $M_{V(u^0)} \times M_{V(u^0)}$ by the Heine-Cantor theorem. This means that for any $\epsilon > 0$ there exists $\delta > 0$ such that if $d_{M\times M}((u^k,u^{k+1}),(u^k,u^k)) = d_M(u^k,u^{k+1}) < \delta$, then

$$\left\|\overline{\mathrm{grad}}V(u^k,u^{k+1})-\mathrm{grad}V(u^k)\right\|_{c^k} = \left\|\overline{\mathrm{grad}}V(u^k,u^{k+1})-\overline{\mathrm{grad}}V(u^k,u^k)\right\|_{c^k} < \epsilon.$$

Since $d_M(u^k,u^{k+1}) \to 0$, given $\epsilon > 0$ there exists K such that for all $k > K$,

$$\left\|\mathrm{grad}V(u^k)\right\|_{c^k} \leq \left\|\overline{\mathrm{grad}}V(u^k,u^{k+1})-\mathrm{grad}V(u^k)\right\|_{c^k} + \left\|\overline{\mathrm{grad}}V(u^k,u^{k+1})\right\|_{c^k} \leq 2\epsilon.$$

This means

$$\lim_{k\to\infty} \mathrm{grad}V(u^k) = 0.$$

Since $M_{V(u^0)}$ is compact, there exists a convergent subsequence $\{u^{k_l}\}$ with limit u^*. Since V is continuously differentiable,

$$\mathrm{grad}V(u^*) = \lim_{l\to\infty} \mathrm{grad}V(u^{k_l}) = 0.$$

□

Remark: In the above proof, we assumed $c^k = u^k$ or $c^k = u^{k+1}$. Although these choices may be desirable for practical purposes, as discussed in the next subsection, one can also make a more general choice. Specifically, if $\phi = \exp$ and c^k, let $\gamma^k(t)$ be the geodesic between u^k and u^{k+1} such that

$$\gamma^k(t) = \exp_{u^k}(tv^k)$$

where $v^k = \exp_{u^k}^{-1}(u^{k+1})$. Then, taking $c^k = \gamma^k(s)$ for some $s \in [0,1]$, uniqueness of geodesics implies that

$$\exp_{c^k}(t\dot{\gamma}^k(s)) = \exp_{u^k}((t+s)v^k).$$

Hence,

$$\exp_{c^k}^{-1}(u^k) = -s\dot{\gamma}^k(s), \qquad \exp_{c^k}^{-1}(u^{k+1}) = (1-s)\dot{\gamma}^k(s),$$

and so, since geodesics are constant speed curves:

$$d(u^k,u^{k+1}) = \|v\|_{u^k} = \|\dot{\gamma}^k(s)\|_{c^k} = \|\exp_{c^k}^{-1}(u^k) - \exp_{c^k}^{-1}(u^{k+1})\|_{c^k}.$$

This means that (4.10) holds in this case. No other arguments in Theorem 4.2 are affected.

4.3.1 Itoh–Abe discrete Riemannian gradient

The Itoh–Abe discrete gradient [18] can be generalized to Riemannian manifolds.

Proposition 4.2. *Given a continuously differentiable energy $V : M \to \mathbb{R}$ and an orthogonal basis $\{E_j\}_{j=1}^{n}$ for $T_{c(u,v)}M$ such that*

$$\phi_c^{-1}(v) - \phi_c^{-1}(u) = \sum_{i=1}^{n} \alpha_i E_i,$$

define $\overline{\mathrm{grad}}_{\mathrm{IA}} V : M \times M \to T_{c(u,v)}M$ by

$$\overline{\mathrm{grad}}_{\mathrm{IA}} V(u, v) = \sum_{j=1}^{n} a_j E_j,$$

where

$$a_j = \begin{cases} \dfrac{V(w_j) - V(w_{j-1})}{\alpha_j}, & \alpha_j \neq 0 \\ g(\mathrm{grad} V(w_{j-1}), d\phi_c\big|_{\eta_{j-1}} E_j), & \alpha_j = 0. \end{cases}$$

$$w_j = \phi_c(\eta_j), \quad \eta_j = \phi_c^{-1}(u) + \sum_{i=1}^{j} \alpha_i E_i.$$

Then, $\overline{\mathrm{grad}}_{\mathrm{IA}} V$ is a discrete Riemannian gradient.

Proof. Continuity of $\overline{\mathrm{grad}}_{\mathrm{IA}} V$ can be seen from the smoothness of the local coordinate frame $\{E_j\}_{j=1}^{n}$ and from the continuity of the $a_j(\alpha_j)$:

$$\begin{aligned} \lim_{\alpha_j \to 0} a_j(\alpha_j) &= \lim_{\alpha_j \to 0} \frac{V\Big(\phi_c\big(\eta_{j-1} + \alpha_j E_j\big)\Big) - V\Big(\phi_c\big(\eta_{j-1}\big)\Big)}{\alpha_j} \\ &= \frac{\mathrm{d}}{\mathrm{d}\alpha_j}\bigg|_{\alpha_j=0} V\Big(\phi_c\big(\eta_{j-1} + \alpha_j E_j\big)\Big) \\ &= \Big\langle \mathrm{d}V\Big(\phi_c\big(\eta_{j-1}\big)\Big), d\phi_c\big|_{\eta_{j-1}} E_j \Big\rangle \\ &= g(\mathrm{grad} V(w_{j-1}), d\phi_c\big|_{\eta_{j-1}} E_j). \end{aligned}$$

Property (4.4) holds since

$$\begin{aligned} g\left(\overline{\text{grad}}_{\text{IA}} V(u,v), \phi_c^{-1}(v) - \phi_c^{-1}(u)\right) &= \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j g(E_i, E_j) \\ &= \sum_{j=1}^{n} V(w_j) - V(w_{j-1}) \\ &= V(w_n) - V(w_0) \\ &= V(v) - V(u). \end{aligned}$$

Furthermore, (4.5) holds since when $v = u$, all $\alpha_j = 0$ and $c(u,v) = u$ so that

$$\overline{\text{grad}}_{\text{IA}} V(u,u) = \sum_{j=1}^{n} g(\text{grad} V(u), E_j) E_j = \text{grad} V(u).$$

□

The map $\overline{\text{grad}}_{\text{IA}} V$ is called the Itoh–Abe discrete Riemannian gradient. For the Itoh–Abe DRG to be a computationally viable option it is important to compute the α_i efficiently. Consider for instance the gradient flow system. Applying the Itoh–Abe DRG to this we get the scheme

$$\begin{aligned} u^{k+1} &= \phi_{c^k}\left(W(u^k, u^{k+1})\right), \\ W(u^k, u^{k+1}) &= \phi_{c^k}^{-1}(u^k) - \tau_k \overline{\text{grad}}_{\text{IA}} V(u^k, u^{k+1}), \end{aligned}$$

meaning

$$\phi_{c^k}^{-1}(u^{k+1}) - \phi_{c^k}^{-1}(u^k) = -\tau_k \overline{\text{grad}}_{\text{IA}} V(u^k, u^{k+1}),$$

and in coordinates

$$\sum_{i=1}^{n} \alpha_i E_i = -\tau_k \sum_{j=1}^{n} \frac{V(w_j) - V(w_{j-1})}{\alpha_j} E_j,$$

so that the α_i are found by solving the n coupled equations

$$\alpha_i = -\tau_k \frac{V(w_i) - V(w_{i-1})}{\alpha_i}.$$

Note that these equations in general are fully implicit in the sense that they require knowledge of the endpoint u^{k+1} since the w_i are dependent on c^k. However, if we take $c^k = u^k$, there is no dependency on the endpoint and all the above equations become scalar, although one must solve them successively. For this choice of c^k we present, as Algorithm 4.1, a procedure for solving the gradient flow problem on a Riemannian manifold with Riemannian metric g using the Itoh–Abe DRG.

Algorithm 4.1 (DRG-OPTIM).

```
Choose $tol > 0$ and $u^0 \in M$. Set $k = 0$.
repeat
    Choose $\tau_k$ and an orthogonal basis $\{E_i^k\}_{i=1}^n$ for $T_{u^k}M$
    $v_0^k = u^k$
    $w_0^k = \phi_{u^k}^{-1}(v_0^k)$
    for $j = 1, ..., n$ do
        Solve $\alpha_j^k = -\tau_k \left( V\left(\phi_{u^k}(w_{j-1}^k + \alpha_j^k E_j^k)\right) - V\left(v_{j-1}^k\right)\right) / \alpha_j^k$
        $w_j^k = w_{j-1}^k + \alpha_j^k E_j^k$
        $v_j^k = \phi_{u^k}(w_j^k)$
    $u^{k+1} = v_n^k$
    $k = k + 1$
until $\left(V(u^k) - V(u^{k-1})\right) / V(u^0) < tol$
```

There is a caveat to this algorithm in that the α_j^k should be easy to compute. For example, it is important that the E_j and ϕ are chosen such that the difference $V(\phi_{u^k}(w_{j-1}^k + \alpha_j^k E_j^k)) - V(v_{j-1}^k)$ is cheap to evaluate. In many cases, M has a natural interpretation as a submanifold of Euclidean space defined locally by constraints $g : \mathbb{R}^m \to \mathbb{R}^n$, $M = \{y \in U \subset \mathbb{R}^m : g(y) = 0\}$. Then, one may find $\{E_j\}_{j=1}^n$ as an orthogonal basis for $\ker g'(c)$ and define ϕ_c implicitly by taking $q = \phi_c(v)$ such that $q - (c + v) \in (T_c M)^\perp$ and $g(q) = 0$, as detailed in [6]. This requires the solution of a nonlinear system of equations for every coordinate update, which is computationally demanding compared to evaluating explicit expressions for $\{E_j\}_{j=1}^n$ and ϕ_c as is possible in special cases, such as those considered in Section 4.4. To compute the α_j^k at each coordinate step one can use any suitable root finder, yet to stay in line with the derivative-free nature of Algorithm 4.1, one may wish to use a solver like the Brent–Dekker algorithm [3]. Also worth noting is that the parallelization procedure used in [22] works for Algorithm 4.1 as well.

4.4 Numerical experiments

This section concerns four applications of DRG methods to gradient flow systems. In each case, we specify all details needed to implement Algorithm 4.1 the manifold M, retraction ϕ, and basis vectors $\{E_k\}$. The first two examples are eigenvalue problems, included to illuminate implementational issues with examples in a familiar setting. We do not claim that our algorithm is competitive with other eigenvalue solvers, but include these examples for the sake of exposition and to have problems with readily available reference solutions. The first of these is a simple Rayleigh quotient minimization problem, where issues of computational efficiency are raised. The second one concerns the Brockett flow on $\mathrm{SO}(m)$, the space of orthogonal $m \times m$ matrices with unit determinant, and serves as an example of optimization on a Lie group. The remaining two problems are examples of manifold-valued image analysis problems concerning Interferometric Synthetic Aperture Radar (InSAR) imaging and Diffusion Tensor Imaging (DTI), respectively. Specifically, the problems concern total variation denoising of images obtained through these techniques [30]. The experiments do not consider the quality of the solution paths, i.e. numerical accuracy. For experiments of this kind, we refer to [5].

All programs used in the following were implemented as MATLAB functions, with critical functions implemented in C using the MATLAB EXecutable (MEX) interface when necessary. The code was executed using MATLAB (2017a release) running on a Mid 2014 MacBook Pro with a four-core 2.5 GHz Intel Core i7 processor and 16 GB of 1600 MHz DDR3 RAM. We used a C language port of the built-in MATLAB function `fzero` for the Brent-Dekker algorithm implementation.

4.4.1 Eigenvalue problems

As an expository example, our first problem consists of finding the smallest eigenvalue/vector pair of a symmetric $m \times m$ matrix A by minimizing its Rayleigh quotient. We shall solve this problem using both the extrinsic and intrinsic view of the $(m-1)$-sphere. In the second example we consider the different approach to the eigenvalue problem proposed by Brockett in [4]. Here, the gradient flow on $\mathrm{SO}(m)$ produces a diagonalizing matrix for a given symmetric matrix.

Eigenvalues via Rayleigh quotient minimization

In our first example, we wish to compute the smallest eigenvalue of a symmetric matrix $A \in \mathbb{R}^{m\times m}$ by minimizing the Rayleigh quotient

$$V(u) = u^T A u$$

with u on the $(m-1)$-sphere S^{m-1}.

Taking the extrinsic view, we regard S^{m-1} as a submanifold in $\mathbb{R}^m$, equipped with the standard Euclidian metric $g(x,y) = x^T y$. In this representation, $T_u S^{m-1}$ is the hyperplane tangent to u, i.e. $T_u S^{m-1} = \{x \in \mathbb{R}^m : x^T u = 0\}$. A natural choice of retraction is

$$\phi_p(x) = \frac{p+x}{\|p+x\|}.$$

There is a difficulty with this ϕ; it does not preserve sparsity, meaning Algorithm 4.1 will be inefficient as discussed above. To see this, consider that at each time step, to find the α_j^k, we must compute the difference

$$V(z_j^k) - V(z_{j-1}^k) = (z_j^k)^T A z_j^k - (z_{j-1}^k)^T A z_{j-1}^k$$

for some $z_{j-1}^k, z_j^k \in S^{m-1}$. We can compute this efficiently if $z_j^k = z_{j-1}^k + \delta$, where δ is sparse. Then,

$$V(z_j^k) - V(z_{j-1}^k) = 2(z_{j-1}^k)^T A\delta + \delta^T A\delta,$$

which is efficient since one may assume Az_{j-1}^k to be precomputed so that the computational cost is limited by the sparsity of δ. In our case, we have

$$z_{j-1}^k = \phi_c(w_{j-1}^k), \qquad z_j^k = \phi_c(w_{j-1}^k + \alpha_j^k E_j).$$

However, with ϕ_c as above, $\delta = \phi_c(w_{j-1}^k + \alpha_j^k E_j) - \phi_c(w_{j-1}^k)$ is non-sparse, and so computing the energy difference is costly.

Next, let us consider the intrinsic view of S^{m-1}, representing it in spherical coordinates $\theta \in \mathbb{R}^{m-1}$ by

$$\begin{aligned}
u_1(\theta) &= \cos(\theta_1),\\
u_r(\theta) &= \cos(\theta_r)\prod_{i=1}^{r-1}\sin(\theta_i), \quad 1 < r < m,\\
u_m(\theta) &= \prod_{i=1}^{m-1}\sin(\theta_i).
\end{aligned}$$

Due to the simple structure of $\mathbb{R}^{m-1}$, we take $\phi_\theta(\eta) = \theta + \eta$. Then, we have

$$u_r(\phi_\theta(\alpha E_l)) = u_r(\theta + \alpha E_l) = \begin{cases} u_r(\theta), & r < l\\ \dfrac{\cos(\theta_l + \alpha)}{\cos(\theta_l)} u_r(\theta), & r = l\\ \dfrac{\sin(\theta_l + \alpha)}{\sin(\theta_l)} u_r(\theta), & r > l. \end{cases}$$

Using this relation, the energy difference after a coordinate update becomes:

$$\begin{aligned}
V(u(\theta + \alpha E_l)) - V(u(\theta)) = {} & 2\kappa_{1l}\sum_{i=1}^{l-1} u_i(\theta)u_l(\theta)A_{il} + 2\kappa_{2l}\sum_{i=1}^{l-1}\sum_{j=l+1}^{m} u_i(\theta)u_j(\theta)A_{ij}\\
& + 2\kappa_{3l}\sum_{j=l+1}^{m} u_l(\theta)u_j(\theta)A_{lj}\\
& + \kappa_{4l}\sum_{i=l+1}^{m}\sum_{j=l+1}^{m} u_i(\theta)u_j(\theta)A_{ij} + \kappa_{5l}u_l(\theta)u_l(\theta)A_{ll},
\end{aligned}$$

with

$$\kappa_{1l} = c_l - 1, \quad \kappa_{2l} = s_l - 1, \quad \kappa_{3l} = s_l c_l - 1, \quad \kappa_{4l} = s_l^2 - 1, \quad \kappa_{5l} = c_l^2 - 1,$$

where

$$c_l = \frac{\cos(\theta_l + \alpha)}{\cos(\theta_l)}, \quad s_l = \frac{\sin(\theta_l + \alpha)}{\sin(\theta_l)}.$$

With prior knowledge of $V(u(\theta))$ (and thus the four partial sums in the difference), evaluating $V(u(\theta+\alpha E_l)) - V(u(\theta))$ amounts to five scalar multiplications and four scalar additions after evaluating the κ_i^l. With correct bookkeeping, new sums can be evaluated from previous sums after coordinate updates, reducing the computational complexity of the algorithm. Although not producing an algorithm competitive with standard eigenvalue solvers, this example demonstrates that the correct choice of coordinates is vital to reducing the computational complexity of the Itoh–Abe DRG method.

Eigenvalues via Brockett flow

Among other things, the article of Brockett [4] discusses how one may find the eigenvalues of a symmetric matrix A by solving the following gradient flow problem on $M = \mathrm{SO}(m)$:

$$\dot{Q} = -Q(DQ^T AQ - Q^T AQD) \tag{4.11}$$

Here, D is a real diagonal matrix with non-repeated entries. It can be shown that $\lim_{t\to\infty} Q = Q^*$, where $(Q^*)^T AQ^* = \Lambda$ is diagonal and hence contains the eigenvalues of A, ordered as the entries of D. Equation (4.11) is the gradient flow of the energy

$$V(Q) = \mathrm{tr}(AQ^T DQ) \tag{4.12}$$

with respect to the trace metric on $\mathrm{SO}(m)$. One can check that $\mathrm{SO}(m)$ is a Lie group [29], with Lie algebra

$$\mathfrak{so}(m) = \{B \in \mathbb{R}^{m\times m} : B^T = -B\}.$$

Also, since $\mathrm{SO}(m)$ is a matrix Lie group, the exponential coincides with the matrix exponential. However, we may consider using some other function as a retraction, such as the Cayley transform $\phi : \mathfrak{so}(m) \to \mathrm{SO}(m)$ given by

$$\phi(B) = (I - B)^{-1}(I + B).$$

Figure 4.1 shows the results of numerical tests with constant time step $\tau_k = 0.1$ and $m = 20$. In the left hand panel, the evolution of the diagonal values of

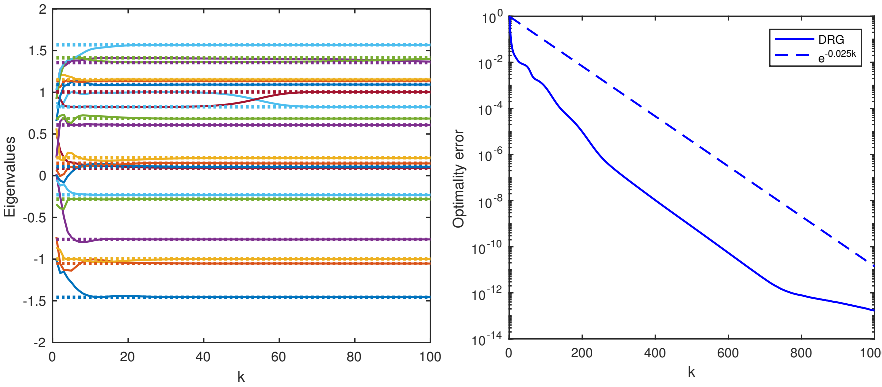


Figure 4.1: Brockett flow with $\tau_k = 0.1$ and 20 eigenvalues. Random initial matrix. Left: Evolution of eigenvalues. Right: Optimality error $(V(u^k) - V^*)/(V(u^0) - V^*)$.

$Q^k AQ^k$ compared to the spectrum of A is shown; it is apparent that the diagonal values converge to the eigenvalues. The right hand panel shows the convergence rate of Algorithm 4.1 to the minimal value V^* as computed with eigenvalues and eigenvectors from MATLAB's `eigen` function. It would appear that the convergence rate is linear, meaning $\|D-(Q^{k+1})^T AQ^{k+1}\| = C\|D-(Q^k)^T AQ^k\|$, with $C < 1$, which corresponds to an exponential reduction in $\|D-(Q^k)^T AQ^k\|$. No noteworthy difference was observed when using the matrix exponential in place of the Cayley transform.

4.4.2 Manifold valued imaging

In the following two examples we will consider problems from manifold valued 2D imaging. We will in both cases work on a product manifold $\mathcal{M} = M^{l\times m}$ consisting of $l \times m$ copies of an underlying data manifold M. An element of M will in this case be called an *atom*, as opposed to the regular term *pixel*. As explained in [20], product manifolds of Riemannian manifolds are again Riemannian manifolds. The tangent spaces of product manifolds have a natural structure as direct sums, with $T_{(u_{11},u_{12},\dots,u_{lm})}\mathcal{M} = \bigoplus_{i,j=1}^{l,m} T_{u_{ij}}M$, which induces a natural Riemannian metric $\mathcal{G} : T\mathcal{M} \times T\mathcal{M} \to \mathbb{R}$ fiberwise as

$$\mathcal{G}_{(u_{11},u_{12},\dots,u_{lm})}((x_{11},\dots,x_{lm}),(y_{11},\dots,y_{lm})) = \sum_{i,j=1}^{l,m} g_{u_{ij}}(x_{ij},y_{ij}).$$

Also, given a retraction $\phi : TM \to M$, one can define a retraction $\Phi : T\mathcal{M} \to \mathcal{M}$ fiberwise as

$$\Phi_{(u_{11},u_{12},\dots,u_{lm})}(x_{11},\dots,x_{lm}) = (\phi_{u_{11}}(x_{11}),\phi_{u_{12}}(x_{12}),\dots,\phi_{u_{lm}}(x_{lm})).$$

Discrete gradients were first used in optimization algorithms for image analysis in [11] and [22]. As an example of a manifold-valued imaging problem, consider Total Variation (TV) denoising of manifold valued images [30], where one wishes to minimize, based on generalizations of the L^β and L^γ norms:

$$\begin{aligned} V(u) = &\frac{1}{\beta}\sum_{i,j=1}^{l,m} \mathrm{d}(u_{ij}, s_{ij})^\beta \\ &+ \lambda\left(\sum_{i,j=1}^{l-1,m} \mathrm{d}(u_{ij}, u_{i+1,j})^\gamma + \sum_{i,j=1}^{l,m-1} \mathrm{d}(u_{ij}, u_{i,j+1})^\gamma\right). \end{aligned} \tag{4.13}$$

Here, $s = (s_{11}, ..., s_{lm}) \in \mathcal{M}$ is the input image, $u = (u_{11}, ..., u_{lm}) \in \mathcal{M}$ is the output image, λ is a regularization strength constant, and d is a metric on M, which we will take to be the geodesic distance induced by g.

InSAR image denoising

We first consider Interferometric Synthetic Aperture Radar (InSAR) imaging, used in earth observation and terrain modelling [24]. In InSAR imaging, terrain elevation is measured by means of phase differences between laser pulses reflected from a surface at different times. Thus, the atoms g_{ij} are elements of $M = S^1$, represented by their phase angles: $-\pi < g_{ij} \leq \pi$. After processing, the phase data is *unwrapped* to form a single, continuous image of displacement data [9]. The natural distance function in this representation is the angular distance

$$\mathrm{d}(\varphi, \theta) = \begin{cases} |\varphi - \theta|, & |\varphi - \theta| \leq \pi \\ 2\pi - |\varphi - \theta|, & |\varphi - \theta| > \pi. \end{cases}$$

Also, $T_\varphi M$ is simply $\mathbb{R}$, and ϕ is given, with $\underset{2\pi}{+}$ denoting addition modulo 2π, as:

$$\phi_\varphi(\theta_\varphi) = (\theta \underset{2\pi}{+} (\varphi + \pi)) - \pi.$$

Figure 4.2 shows the result of applying TV denoising to an InSAR image of a slope of Mt. Vesuvius, Italy, with $\beta = 2$. The left column shows the phase data, while the right hand side shows the phase unwrapped data. The input image was taken from [23]. It is evident that the algorithm is successful in removing noise. Computation time was 0.1 seconds per iteration on a 150×150 image. A logarithmic plot showing convergence in terms of $(V(u^k) - V^*)/(V(u^0) - V^*)$ is shown in Figure 4.3, where V^* is a near-optimal value for V, obtained by iterating until $V(u^{k+1}) - V(u^k) \leq 10^{-15}$. The plot shows the behaviour of Algorithm 4.1 with constant time steps $\tau_k = \tau_0 = 0.002$ and an ad-hoc adaptive

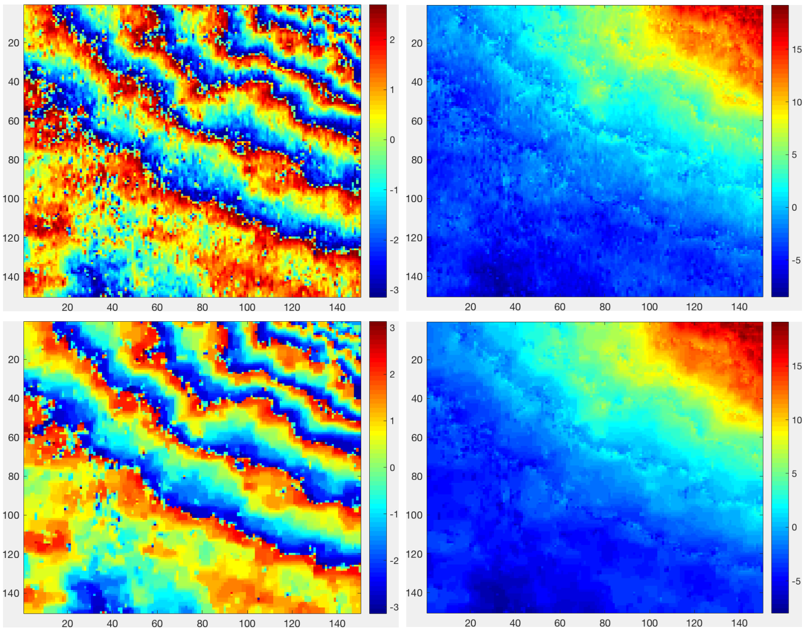


Figure 4.2: Left column: Interferogram. Right column: Phase unwrapped image. Top row: Original image. Bottom row: L2 fidelity denoising, $\lambda = 0.3$.

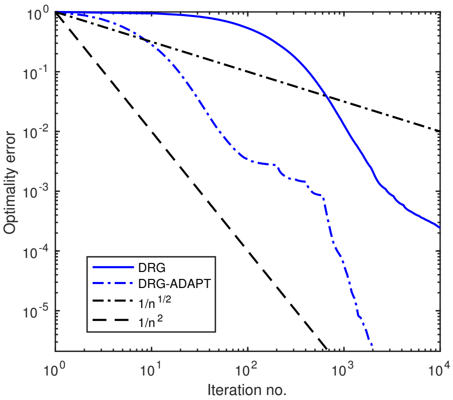


Figure 4.3: Logarithmic plot of optimality error $(V(u^k) - V^*)/(V(u^0) - V^*)$.

method with $\tau_0 = 0.005$ where τ_k is halved each 200 iterations; for each of these strategies a separate V^* was found since they did not produce convergence to the same minimizer. The reason for the different minimizers is that the TV functional, and thus the minimization problem, is non-convex in S^1 [27]. We can observe that the convergence speed varies between $\mathcal{O}(1/k)$ and $\mathcal{O}(1/k^2)$, with faster convergence for the ad-hoc adaptive method. The reason for this sublinear convergence as compared to the linear convergence observed in the Brockett flow case may be the non-convexity.

DTI image denoising

Diffusion Tensor Imaging (DTI) is a medical imaging technique where the goal is to make spatial samples of the tensor specifying the diffusion rates of water in biological tissue. The tensor is assumed to be, at each point (i,j), represented by a matrix $A_{ij} \in \mathrm{Sym}^+(3)$, the space of 3×3 symmetric positive definite (SPD) matrices. Experimental measurements of DTI data are, as with other MRI techniques, contaminated by Rician noise [12], which one may attempt to remove by minimizing (4.13) with an appropriate choice of Riemannian structure on $\mathcal{M} = \mathrm{Sym}^+(3)^{m\times l}$.

As above, since the manifold we are working on is a product manifold, it suffices to define the Riemannian structure on $\mathrm{Sym}^+(3)$. First off, one should note that $T_A\mathrm{Sym}^+(3)$ can be identified with $\mathrm{Sym}(3)$, the space of symmetric 3×3 matrices [19]. In [30], the authors consider equipping $\mathrm{Sym}^+(3)$ with the affine invariant Riemannian metric given pointwise as

$$g_A(X,Y) = \mathrm{tr}(A^{-\frac{1}{2}}XA^{-1}YA^{-\frac{1}{2}}),$$

and for purposes of comparison, so shall we. The space $\mathrm{Sym}^+(3)$ equipped with this metric is a Cartan-Hadamard manifold [19], and thus is complete, meaning that Theorem 4.2 holds. This metric induces the explicitly computable geodesic distance

$$d(A,B) = \sqrt{\sum_{i=1}^{3} \log(\kappa_i)^2}$$

on $\mathrm{Sym}^+(3)$, where κ_i are the eigenvalues of $A^{-\frac{1}{2}}BA^{-\frac{1}{2}}$. Furthermore, the metric induces a Riemannian exponential given by

$$\exp_A(Y) = A^{1/2}\mathrm{e}^{A^{-1/2}YA^{-1/2}}A^{1/2}$$

where e denotes the matrix exponential, and $A^{1/2}$ is the matrix square root of A. We could choose the retraction as $\phi = \exp$, but there are less computationally expensive options that do not involve computing matrix exponentials. More specifically, we will make use of the second-order approximation of the exponential,

$$\phi_A(Y) = A + Y + \frac{1}{2}YA^{-1}Y.$$

While a first-order expansion is also a retraction, there is no guarantee that $A + Y \in \mathrm{Sym}^+(3)$, whereas the second-order expansion, which can be written on the form

$$\phi_A(Y) = \frac{1}{2}A + \frac{1}{2}(A^{\frac{1}{2}} + A^{-\frac{1}{2}}Y)^T(A^{\frac{1}{2}} + A^{-\frac{1}{2}}Y),$$

is clearly symmetric positive definite since A is so. Note that using a sparse basis E_{ij} (in our example we use $E_{ij} = e_i e_j^T + e_j e_i^T$) for the space Sym(3), evaluating $\phi_A(X + \alpha E_{ij})$ amounts to, at most, four scalar updates when $\phi_A(X)$ and A^{-1} is known, as is possible with proper bookkeeping in the software implementation. Also, since all matrices involved are 3×3 SPD matrices, one may find eigenvalues and eigenvectors directly, thus allowing for fast computations of matrix square roots and, consequently, geodesic distances.

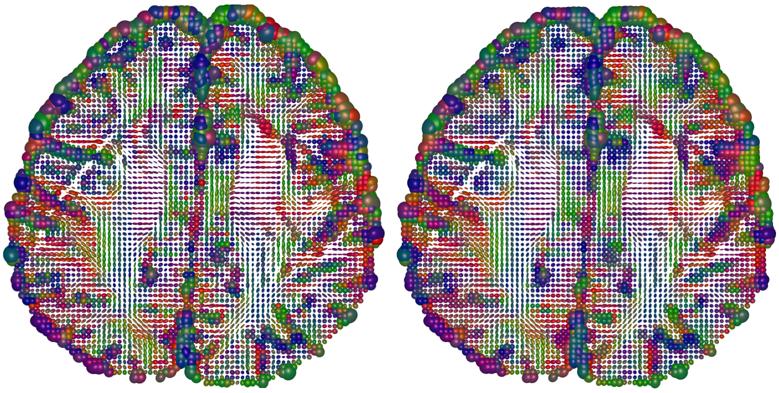


Figure 4.4: DTI scan, axial slice. Left: Noisy image. Right: Denoised with $\beta = 2$, $\lambda = 0.05$.

Figure 4.4 shows an example of denoising DTI images using the TV regularizer. The data is taken from the publicly available Camino data set [8]. The DTI tensor has been calculated from underlying data using linear least-squares fitting, and is subject to Rician noise (left hand side), which is mitigated by TV denoising (right hand side). The denoising procedure took about 7 seconds for 57 iterations, on a 72×73 image. The algorithm was stopped when the relative change in energy, $(V(u^0) - V(u^k))/V(u^0)$ dropped below 10^{-5}. Each atom $A \in \mathrm{Sym}^+(3)$ is visualized by an ellipsoid with the eigenvectors of A as principal semi-axes, scaled by the corresponding eigenvalues. The colors are coded to correspond to the principal direction of the major axis, with red denoting left-right orientation, green anterior-posterior and blue inferior-superior. Figure 4.5 shows the convergence behaviour of Algorithm 4.1, with three different time steps: $\tau = 0.05$, $\tau = 0.01$ and a mixed strategy of using $\tau = 0.05$ for 12 steps, then changing to $\tau = 0.01$. Also, baseline rates of $1/k^2$ and $1/k$ are shown. It is

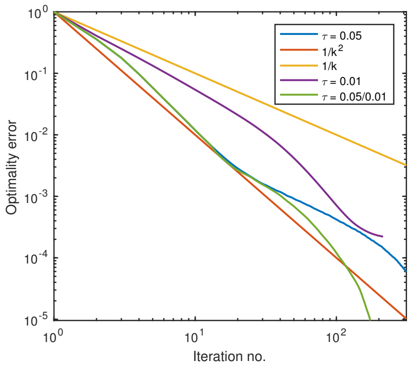


Figure 4.5: Logarithmic plot of optimality error.

apparent that the choice of time step has great impact on the convergence rate, and that simply changing the time step from $\tau = 0.05$ to $\tau = 0.01$ is effective in speeding up convergence. This would suggest that time step adaptivity is a promising route for acceleration of these methods.

4.5 Conclusion and outlook

We have extended discrete gradient methods to Riemannian manifolds, and shown how they may be applied to gradient flows. The Itoh–Abe discrete gradient has been formulated in a manifold setting; this is, to the best of our knowledge, the first time this has been done. In particular, we have used the Itoh–Abe DRG on gradient systems to produce a derivative-free optimization algorithm on Riemannian manifolds. This optimization algorithm has been proven to converge under reasonable conditions, and shows promise when applied to the problem of denoising manifold valued images using the total variation approach of [30].

As with the algorithm in the Euclidian case, there are open questions. The first question is which convergence rate estimates can be made; one should especially consider the linear convergence exhibited in the Brockett flow problem, and the rate observed in Figure 4.5 which approaches $1/k^2$. A second question is how to formulate a rule for choosing step sizes so as to accelerate convergence toward minimizers. There is also the question of how the DRG methods perform as ODE solvers for dissipative problems on Riemannian manifolds; in particular, convergence properties, stability, and convergence order. The above discussion is geared toward optimization applications due to the availability of optimization problems, but it would be of interest to see how the methods work as ODE solvers in their own right similar to the analysis and experiments done in [5].

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[2] R. L. Adler, J.-P. Dedieu, J. Y. Margulies, M. Martens, and M. Shub. Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA J. Numer. Anal.*, 22(3):359–390, 2002.

[3] R. P. Brent. An algorithm with guaranteed convergence for finding a zero of a function. *Comput. J.*, 14(4):422–425, 1971.

[4] R. W. Brockett. Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems. In *IEEE Decis. Contr. P.*, pages 799–803. IEEE, 1988.

[5] E. Celledoni, S. Eidnes, B. Owren, and T. Ringholm. Energy-preserving methods on Riemannian manifolds. *Math. Comp.*, 89(322):699–716, 2020.

[6] E. Celledoni and B. Owren. A class of intrinsic schemes for orthogonal integration. *SIAM J. Numer. Anal.*, 40(6):2069–2084 (2003), 2002.

[7] E. Celledoni and B. Owren. Preserving first integrals with symmetric Lie group methods. *Discrete Cont. Dyn. S.*, 34(3):977–990, 2014.

[8] P. Cook, Y. Bai, S. Nedjati-Gilani, K. Seunarine, M. Hall, G. Parker, and D. Alexander. Camino: open-source diffusion-MRI reconstruction and processing. In *Proc. 14th Sci. Meeting of ISMRM*, volume 2759. Seattle WA, USA, 2006.

[9] R. M. Goldstein, H. A. Zebker, and C. L. Werner. Satellite radar interferometry: Two-dimensional phase unwrapping. *Radio Sci.*, 23(4):713–720, 1988.

[10] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[11] V. Grimm, R. I. McLachlan, D. I. McLaren, G. Quispel, and C. Schönlieb. Discrete gradient methods for solving variational image regularisation models. *J. Phys. A: Math. Theor.*, 50(29):295201, 2017.

[12] H. Gudbjartsson and S. Patz. The Rician distribution of noisy MRI data. *Magn. Reson. Med.*, 34(6):910–914, 1995.

[13] E. Hairer and C. Lubich. Energy-diminishing integration of gradient systems. *IMA J. Numer. Anal.*, 34(2):452–461, 2013.

[14] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[15] E. Hairer and G. Wanner. *Solving ordinary differential equations. II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1996. Stiff and differential-algebraic problems.

[16] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, 25(1):35–61, 1983.

[17] A. Humphries and A. Stuart. Runge–Kutta methods for dissipative and gradient dynamical systems. *SIAM J. Numer. Anal.*, 31(5):1452–1485, 1994.

[18] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 76(1):85–102, 1988.

[19] S. Lang. *Fundamentals of differential geometry*, volume 191. Springer Science & Business Media, 2012.

[20] J. M. Lee. *Riemannian manifolds: an introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

[21] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *Philos. T. R. Soc. A*, 357(1754):1021–1045, 1999.

[22] T. Ringholm, J. Lazić, and C.-B. Schönlieb. Variational image regularization with Euler's elastica using a discrete gradient scheme. *SIAM J. Imaging Sci.*, 11(4):2665–2691, 2018.

[23] F. Rocca, C. Prati, and A. Ferretti. An overview of ERS-SAR interferometry. In *ERS Symp. Space Serv. Env.*, 1997.

[24] P. A. Rosen, S. Hensley, I. R. Joughin, F. K. Li, S. N. Madsen, E. Rodriguez, and R. M. Goldstein. Synthetic aperture radar interferometry. *P. IEEE*, 88(3):333–382, 2000.

[25] S. Sato, T. Matsuo, H. Suzuki, and D. Furihata. A Lyapunov-type theorem for dissipative numerical integrators with adaptive time-stepping. *SIAM J. Numer. Anal.*, 53(6):2505–2518, 2015.

[26] M. Shub. Some remarks on dynamical systems and numerical analysis. *P. VII ELAM.*, pages 69–92, 1986.

[27] E. Strekalovskiy and D. Cremers. Total variation for cyclic structures: Convex relaxation and efficient minimization. In *Proc. Cvpr. IEEE*, pages 1905–1911. IEEE Computer Society, 2011.

[28] C. Udriste. *Convex functions and optimization methods on Riemannian manifolds*, volume 297. Springer Science & Business Media, 1994.

[29] F. W. Warner. *Foundations of differentiable manifolds and Lie groups*, volume 94. Springer Science & Business Media, 2013.

[30] A. Weinmann, L. Demaret, and M. Storath. Total variation regularization for manifold-valued data. *SIAM J. Imaging Sci.*, 7(4):2226–2257, 2014.

Order theory for discrete gradient methods

Sølve Eidnes

Submitted

Order theory for discrete gradient methods

Abstract. We present a subclass of the discrete gradient methods, which are integrators designed to preserve invariants of ordinary differential equations. From a formal series expansion of the methods, we derive conditions for arbitrarily high order. We devote considerable space to the average vector field discrete gradient, from which we get P-series methods in the general case, and B-series methods for canonical Hamiltonian systems. Higher order schemes are presented and applied to the Hénon–Heiles system and a Lotka–Volterra system.

5.1 Energy preservation and discrete gradient methods

For an ordinary differential equation (ODE)

$$\dot{x} = f(x), \quad x \in \mathbb{R}^d, \quad f : \mathbb{R}^d \to \mathbb{R}^d, \tag{5.1}$$

a first integral, or invariant, is a function $H : \mathbb{R}^d \to \mathbb{R}$ such that $H(x(t)) = H(x(t_0))$ along the solution curves of (5.1). If we can write

$$f(x) = S(x)\nabla H(x), \tag{5.2}$$

where $S(x) : \mathbb{R}^{d\times d} \to \mathbb{R}^d$ is a skew-symmetric matrix, then (5.1) preserves H: this follows from the skew-symmetry of $S(x)$, which yields

$$\frac{\mathrm{d}}{\mathrm{d}t}H(x) = \nabla H(x)^T \dot{x} = \nabla H(x)^T S(x)\nabla H(x) = 0. \tag{5.3}$$

The converse is also true: McLachlan et al. showed in [20] that, whenever (5.1) has a first integral H, there exists a skew-symmetric matrix $S(x)$, bounded near every non-degenerate critical point of H, such that (5.1) can be written on what is called the *skew-gradient form*:

$$\dot{x} = S(x)\nabla H(x). \tag{5.4}$$

The proof provided in [20] for this is based on presenting a general form of one such $S(x)$, the so-called default formula

$$S(x) = \frac{f(x)\nabla H(x)^T - \nabla H(x) f(x)^T}{\nabla H(x)^T \nabla H(x)}. \tag{5.5}$$

Unless $d = 2$, this is generally not a unique choice of $S(x)$, as e.g.

$$S(x) = \frac{f(x)g(x)^T - g(x)f(x)^T}{g(x)^T \nabla H(x)}$$

will satisfy (5.2) for any non-vanishing function $g : \mathbb{R}^d \to \mathbb{R}^d$. Many ODEs with first integrals have a well-known skew-gradient form (5.4). This includes Poisson systems, and the important class consisting of canonical Hamiltonian ODEs. For the latter, S will be constant, so that we may write

$$\dot{x} = S\nabla H(x). \tag{5.6}$$

A numerical integrator preserving a first integral H exactly is called an integral-preserving, or *energy-preserving*, method. Starting in the late 1970s, a few energy-preserving methods were proposed which relied on some discrete analogue of the property (5.3), see e.g. [4,15–17]. Most prominent among these is the class of methods called discrete gradient methods, defined formally by Gonzalez in [11] and given their current name in [20].

Given the first integral H, a discrete gradient $\overline{\nabla} H : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ is a function satisfying the conditions

$$\overline{\nabla} H(x,y)^{\mathrm{T}}(y-x) = H(y) - H(x), \tag{5.7}$$

$$\overline{\nabla} H(x,x) = \nabla H(x), \tag{5.8}$$

for all $x, y \in \mathbb{R}^d$. Introducing also the discrete approximation $\overline{S}(x,y,h)$ to $S(x)$, skew-symmetric and satisfying $\overline{S}(x,x,0) = S(x)$, the corresponding discrete gradient method is given by

$$\frac{\hat{x}-x}{h} = \overline{S}(x,\hat{x},h)\overline{\nabla} H(x,\hat{x}). \tag{5.9}$$

This scheme satisfies a discrete analogue to (5.3):

$$H(\hat{x}) - H(x) = h\overline{\nabla} H(x,\hat{x})^T \overline{S}(x,\hat{x},h)\overline{\nabla} H(x,\hat{x}) = 0.$$

We say that (5.9) is consistent to the skew-gradient system (5.4), since $\overline{S}(x,\hat{x},h)$ is a consistent approximation of $S(x)$ and $\overline{\nabla} H(x,\hat{x})$ is a consistent approximation of $\nabla H(x)$.

If $d \geq 2$, there are in general infinitely many functions satisfying (5.7)–(5.8). Many explicit definitions of concrete discrete gradients have been suggested, and we will discuss the most prominent among them in Section 5.2.1. One of these is the *average vector field (AVF) discrete gradient*, first introduced in [14] and sometimes called the mean value discrete gradient [20]. For a given H, it is given by the average of ∇H on the segment $[x,y]$:

$$\overline{\nabla}_{\mathrm{AVF}} H(x,y) = \int_0^1 \nabla H((1-\xi)x + \xi y)\,\mathrm{d}\xi. \tag{5.10}$$

When applied to the constant S system (5.6), the discrete gradient method with $\overline{S}(x,y,h) = S$ and $\overline{\nabla} H = \overline{\nabla}_{\mathrm{AVF}} H$ coincides with the scheme

$$\frac{\hat{x}-x}{h} = \int_0^1 f((1-\xi)x + \xi\hat{x})\,\mathrm{d}\xi. \tag{5.11}$$

This is sometimes viewed as a method by itself, applicable to any system (5.1), in which case it is called the average vector field (AVF) method [26]. This was shown in [2] to be a B-series method.

As pointed out in [20], the discrete gradient is restricted by its definition to be at best a second order approximation to point values of ∇H. In much of the literature on discrete gradient methods, see e.g. [11, 13], the approximation $\overline{S}$ is defined as being independent of h. In that case, the discrete gradient scheme (5.9) can at best guarantee second order convergence towards the exact solution. Over the last two decades, there have been published some notable papers on higher order discrete gradient methods. McLaren and Quispel were first out with their bootstrapping technique derived in [21, 22]. Given any discrete gradient $\overline{\nabla} H$ and an approximation to $S(x)$ given by $\overline{S}(x, y, h)$, they compare the Taylor expansion of the corresponding discrete gradient scheme to that of the exact solution, and thus find a new approximation $\tilde{S}(x, y, h)$ to $S(x)$ which yields higher order. This quickly becomes a very involved procedure, but by using a symmetric discrete gradient, they derive fourth order methods. A downside of this method is that the schemes of order higher than two require the calculation of tensors of order three or higher at every time step.

A fourth order generalization of the AVF method is proposed by the same authors in [26]. This can be viewed as a fourth-order discrete gradient method for all skew-gradient systems where S is constant. Also worth mentioning in this setting is the collocation-like method introduced by Hairer [12] and then generalized to Poisson systems by Cohen and Hairer [5]. This is a multi-stage extension of the AVF discrete gradient method. To get higher than second order, more than one stage is required. In that case the method is not a discrete gradient method, although it is energy-preserving.

Norton et al. show in [24] that linear projection methods can be viewed as a class of discrete gradient methods for skew-gradient systems with $S(x)$ given by the default formula (5.5). In connection to this, Norton and Quispel suggest in [25] the class of approximations to (5.5) given by

$$\overline{S}(x, y, h) = \frac{\tilde{f}(x, y, h)\breve{g}(x, y, h)^T - \breve{g}(x, y, h)\tilde{f}(x, y, h)^T}{\hat{g}(x, y, h)^T \breve{g}(x, y, h)}, \qquad (5.12)$$

where $\tilde{f}(x, y, h)$ is a consistent approximation to $f(x)$, and $\tilde{g}(x, y, h)$, $\hat{g}(x, y, h)$ and $\breve{g}(x, y, h)$ are all consistent approximations to $\nabla H(x)$. The corresponding discrete gradient method then inherits the order of the method $\hat{x} = x + h\tilde{f}(x, \hat{x}, h)$.

To the best of our knowledge, no one has so far suggested higher than fourth order discrete gradient methods for a general skew-gradient system (5.4). Furthermore, for this general case, all discrete gradient methods suggested of higher than second order involve tensors of order three or higher. Our aim with this paper is to remedy this. Largely inspired by the above mentioned references, especially [21,22,26], we present here a general form giving a class of approximations $\overline{S}(x,y,h)$ to any $S(x)$ in (5.4), with corresponding conditions for achieving an arbitrary order of the discrete gradient method (5.9). We do this step by step. In the next chapter, we derive some useful properties of a general discrete gradient and discuss the most common specific discrete gradients. Then we consider the AVF method and use order theory for B-series methods to obtain a generalization of this, with corresponding order conditions. In Chapter 5.4, we build on this to develop higher order discrete gradient methods for a general skew-gradient system, using the AVF discrete gradient. Then, in Chapter 5.5, we generalize this further to allow for a free choice of the discrete gradient, thus arriving at the general form $\overline{S}(x,y,h)$ mentioned above, and a formal series expansion of the corresponding discrete gradient methods. We present several examples of higher order schemes for the different cases, and conclude the paper with some numerical experiments.

5.2 A preliminary analysis of discrete gradients

To simplify notation in the following derivations, we define $g := \nabla H$. Furthermore, we suppress the first argument of $\overline{\nabla} H$ and define $\bar{g}(y) := \overline{\nabla} H(x,y)$. We use Einstein summation convention and write $\bar{g}(y)^i_j := \frac{\partial \bar{g}(y)^i}{\partial y^j}$ and so forth. Taylor expanding $\bar{g}(y)$ around x, we get

$$\begin{aligned}\bar{g}(y)^i &= \bar{g}(x)^i + \bar{g}(x)^i_j(y^j - x^j) + \frac{1}{2}\bar{g}(x)^i_{jk}(y^j - x^j)(y^k - x^k)\\ &\quad + \frac{1}{6}\bar{g}(x)^i_{jkl}(y^j - x^j)(y^k - x^k)(y^l - x^l) + \mathcal{O}(|y-x|^4),\end{aligned} \tag{5.13}$$

or

$$\bar{g}(y) = \sum_{\kappa=0}^{\infty} \frac{1}{\kappa!}\bar{g}^{(\kappa)}(x)(y-x)^{\kappa}. \tag{5.14}$$

By the consistency criterion (5.8), we have $\bar{g}(x) = g(x)$. However, if we require the discrete gradient to be a differentiable function in its second argument, (5.8) follows directly from (5.7). To see this, we write (5.7) as

$$H(y) - H(x) = \bar{g}(y)_i(y^i - x^i). \tag{5.15}$$

Differentiating this with respect to y^j, we get

$$g(y)_j = H(y)_j = \bar{g}(y)_{i,j}(y^i - x^i) + \bar{g}(y)_j, \tag{5.16}$$

where $H_j = \frac{\partial H}{\partial y^j}$ and $\bar{g}(y)_{i,j} = \frac{\partial \bar{g}(y)_i}{\partial y^j}$. The case $y = x$ immediately gives $g(x)_j = \bar{g}(x)_j$, or (5.8). Assuming further that $\overline{\nabla} H \in C^2(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$, we can differentiate once more to get

$$g(y)_{j,k} = H(y)_{jk} = \bar{g}(y)_{i,jk}(y^i - x^i) + \bar{g}(y)_{j,k} + \bar{g}(y)_{k,j}, \tag{5.17}$$

which means that

$$g(x)_{j,k} = H(x)_{jk} = \bar{g}(x)_{j,k} + \bar{g}(x)_{k,j},$$

or

$$\nabla^2 H(x) = D_2\overline{\nabla} H(x,x) + (D_2\overline{\nabla} H(x,x))^{\mathrm{T}}, \tag{5.18}$$

where $\nabla^2 H := D\nabla H$ denotes the Hessian of H, and $D_2\overline{\nabla} H$ denotes the Jacobian of $\overline{\nabla} H$ with respect to its second argument.

Lemma 5.1. *If the discrete gradient $\overline{\nabla} H$ is symmetric, i.e. $\overline{\nabla} H(x,y) = \overline{\nabla} H(y,x)$ for all $x, y \in \mathbb{R}^d$, then*

$$D_2\overline{\nabla} H(x,x) = \frac{1}{2}\nabla^2 H(x). \tag{5.19}$$

Proof. Disclosing the suppressed argument x in (5.16), we have

$$g(y)_j = \frac{\partial}{\partial y^j}(\bar{g}(x,y)_i)(y^i - x^i) + \bar{g}(x,y)_j,$$

which we can differentiate by x^k to get

$$0 = \frac{\partial^2}{\partial x^k \partial y^j}(\bar{g}(x,y)_i)(y^i - x^i) - \frac{\partial}{\partial y^j}\bar{g}(x,y)_k + \frac{\partial}{\partial x^k}\bar{g}(x,y)_j.$$

If $\overline{\nabla} H$ is symmetric,

$$\frac{\partial}{\partial x^k}\bar{g}(x,y)_j = \frac{\partial}{\partial x^k}\bar{g}(y,x)_j.$$

Thus, for $y = x$ we get $\bar{g}(x)_{k,j} = \bar{g}(x)_{j,k}$, or $(D_2\overline{\nabla} H(x,x))^{\mathrm{T}} = D_2\overline{\nabla} H(x,x)$. Inserting that in (5.18), we obtain (5.19). □

Definition 5.1. Given a discrete gradient $\overline{\nabla} H \in C^1(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$, we define the function $Q : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{d\times d}$ by

$$Q(x,y) := \frac{1}{2}\Big((D_2\overline{\nabla} H(x,y))^T - D_2\overline{\nabla} H(x,y)\Big). \tag{5.20}$$

Note that $Q(x,y)$ is a skew-symmetric matrix. From (5.18), we see that $Q(x,x) = \frac{1}{2}\nabla^2 H(x) - D_2\overline{\nabla} H(x,x)$. Differentiating (5.20) with respect to the second argument and setting $y = x$, we obtain

$$(D_2 Q(x,x))_{jkl} = \frac{1}{2}\bar{g}(x)_{k,jl} - \frac{1}{2}\bar{g}(x)_{j,kl}.$$

Similarly, differentiating (5.17) with respect to the second argument and setting $y = x$, we obtain

$$g(x)_{j,kl} = \bar{g}(x)_{j,kl} + \bar{g}(x)_{k,jl} + \bar{g}(x)_{l,jk}.$$

Using these results, we get that, for any $v \in \mathbb{R}^d$,

$$\begin{aligned}
(D_2Q(x,x)(v,v))_j &= (D_2Q(x,x))_{jkl}v^k v^l \\
&= \frac{1}{2}\bar{g}(x)_{k,jl}v^k v^l - \frac{1}{2}\bar{g}(x)_{j,kl}v^k v^l \\
&= \frac{1}{4}\bar{g}(x)_{k,jl}v^k v^l + \frac{1}{4}\bar{g}(x)_{l,jk}v^k v^l \\
&\quad + \frac{1}{4}\bar{g}(x)_{j,kl}v^k v^l - \frac{3}{4}\bar{g}(x)_{j,kl}v^k v^l \\
&= \frac{1}{4}g(x)_{j,kl}v^k v^l - \frac{3}{4}\bar{g}(x)_{j,kl}v^k v^l,
\end{aligned}$$

or

$$D_2Q(x,x)(v,v) = \frac{1}{4}D^2\nabla H(x)(v,v) - \frac{3}{4}D_2^2\overline{\nabla}H(x,x)(v,v). \tag{5.21}$$

Continuing in this manner, we get the following general result, which will be useful when developing higher order discrete gradient methods.

Lemma 5.2. *For a discrete gradient $\overline{\nabla}H \in C^p(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R})$ and the corresponding Q given by* (5.20)*,*

$$D_2^{\kappa}\overline{\nabla}H(x,x)v^{\kappa} = \frac{1}{\kappa+1}D^{\kappa}\nabla H(x)v^{\kappa} - \frac{2\kappa}{\kappa+1}D_2^{\kappa-1}Q(x,x)v^{\kappa},$$

for any $\kappa \in [1,p]$, $v \in \mathbb{R}^d$.

Proof. Differentiating (5.17) $\kappa - 1$ times by y and setting $y = x$, we find that the κ-th derivatives of $g(x)$ can be expressed by the κ-th derivatives of $\bar{g}(x)$ through the relation

$$g(x)_{j,I} = \bar{g}(x)_{j,I} + \sum_{m=1}^{\kappa}\bar{g}(x)_{i_m,\{j,I_m\}}, \qquad \text{for all } j, I, \kappa, \tag{5.22}$$

where $I = \{i_1, i_2, \ldots, i_\kappa\}$ is an ordered set of κ indices, and $I_m = I \setminus \{i_m\} = \{i_1, i_2, \ldots i_{m-1}, i_{m+1}, \ldots, i_\kappa\}$, i.e. I with the m-th element excluded. Similarly, by continued differentiation of (5.20), we obtain

$$(D_2^{\kappa-1}Q(x,x))_{j,I} = \frac{1}{2}\bar{g}(x)_{i_1,\{j,I_1\}} - \frac{1}{2}\bar{g}(x)_{j,I}.$$

Thus

$$\begin{aligned}
(D_2^{\kappa-1}Q(x,x)v^{\kappa})_j &= (D_2^{\kappa-1}Q(x,x))_{j,I}v^I \\
&= \frac{1}{2}\bar{g}(x)_{i_1,\{j,I_1\}}v^{\{j,I_1\}} - \frac{1}{2}\bar{g}(x)_{j,I}v^I \\
&= \frac{1}{2\kappa}\sum_{m=1}^{\kappa}\bar{g}(x)_{i_m,\{j,I_m\}}v^{\{j,I_m\}} \\
&\quad + \frac{1}{2\kappa}\bar{g}(x)_{j,I}v^I - \frac{1}{2\kappa}\bar{g}(x)_{j,I}v^I - \frac{1}{2}\bar{g}(x)_{j,I}v^I \\
&= \frac{1}{2\kappa}g(x)_{j,I}v^I - (\frac{1}{2\kappa}+\frac{1}{2})\bar{g}(x)_{j,I}v^I \\
&= \frac{1}{2\kappa}g(x)_{j,I}v^I - \frac{\kappa+1}{2\kappa}\bar{g}(x)_{j,I}v^I.
\end{aligned}$$

□

5.2.1 A review of explicitly defined discrete gradients

While introducing the discrete gradient methods in [11], Gonzalez also gave an example of a discrete gradient satisfying (5.7)–(5.8): the *midpoint discrete gradient* is given by

$$\overline{\nabla}_{\mathrm{M}}H(x,y) := \nabla H\left(\frac{x+y}{2}\right) + \frac{H(y)-H(x)-\nabla H\left(\frac{x+y}{2}\right)^T(y-x)}{(y-x)^T(y-x)}\big(y-x\big).$$

Even when H is analytical, this discrete gradient is often not; the second order partial derivatives are in general singular in $y = x$. For that reason, it is not suited for achieving higher order methods by the techniques we consider in this paper.

The *Itoh–Abe discrete gradient*, introduced in [15], notably does not require evaluation of the gradient. This discrete gradient, which has also been called the coordinate increment discrete gradient [20], is defined by

$$\overline{\nabla}_{\mathrm{IA}}H(x,y) := \sum_{j=1}^{d}\alpha_j e_j, \tag{5.23}$$

where e_j is the j^{th} canonical unit vector and

$$\begin{aligned}
\alpha_j &= \begin{cases} \dfrac{H(w_j)-H(w_{j-1})}{y^j-x^j} & \text{if } y^j \neq x^j, \\ \frac{\partial H}{\partial x^j}(w_{j-1}) & \text{if } y^j = x^j, \end{cases} \\
w_j &= \textstyle\sum_{i=1}^{j} y^i e_i + \sum_{i=j+1}^{n} x^i e_i.
\end{aligned}$$

While the other discrete gradients we consider in this paper are symmetric and thus second order approximations to ∇H, the Itoh–Abe discrete gradient is only of first order. However, a second order discrete gradient, which we call the *symmetrized Itoh–Abe (SIA) discrete gradient*, is given by

$$\overline{\nabla}_{\mathrm{SIA}}H(x,y) := \frac{1}{2}\left(\overline{\nabla}_{\mathrm{IA}}H(x,y) + \overline{\nabla}_{\mathrm{IA}}H(y,x)\right). \tag{5.24}$$

Furihata presented the discrete variational derivative method for a class of partial differential equations (PDEs) in [9], a method which has been developed further by Furihata, Matsuo and co-authors in a series of papers, e.g. [19, 27], as well as the monograph [10]. As shown in [7], these schemes can also be obtained by semi-discretizing the PDE in space and then applying a discrete gradient method on the resulting system of ODEs. The specific discrete gradient that gives the schemes of Furihata and co-authors is defined for a class of invariants that includes all polynomial functions:

Definition 5.2. Assume that we can write the first integral as

$$H(x) = \sum_l c_l \prod_{j=1}^{d} f_j^l(x^j), \tag{5.25}$$

for functions $f_j^l : \mathbb{R} \to \mathbb{R}$. The *Furihata discrete gradient* $\overline{\nabla}_{\mathrm{F}}H(x,y)$ is defined by

$$\overline{\nabla}_{\mathrm{F}}H(x,y) := \sum_{j=1}^{d} \alpha_j e_j, \tag{5.26}$$

where e_j is the j^{th} canonical unit vector and

$$\alpha_j = \begin{cases} \sum_l \frac{c_l}{2} \frac{f_j^l(y^j)-f_j^l(x^j)}{y^j-x^j} \left(\prod_{k=1}^{j-1} f_k^l(x^k) + \prod_{k=1}^{j-1} f_k^l(y^k)\right) \prod_{k=j+1}^{d} \frac{f_k^l(x^k)+f_k^l(y^k)}{2} & \text{if } y^j \neq x^j, \\ \sum_l \frac{c_l}{2} \frac{\mathrm{d}f_j^l(x^j)}{\mathrm{d}x^j} \left(\prod_{k=1}^{j-1} f_k^l(x^k) + \prod_{k=1}^{j-1} f_k^l(y^k)\right) \prod_{k=j+1}^{d} \frac{f_k^l(x^k)+f_k^l(y^k)}{2} & \text{if } y^j = x^j. \end{cases}$$

Lastly we consider the AVF discrete gradient (5.10), which distinguishes itself from the others in a number of ways.

Lemma 5.3. *The $Q(x,y)$ corresponding to the AVF discrete gradient is the zero matrix, since $(D_2\overline{\nabla}_{AVF}H(x,y))^T = D_2\overline{\nabla}_{AVF}H(x,y)$.*

Proof. For $\bar{g}(y) := \overline{\nabla}_{\mathrm{AVF}}H(x,y)$, we have

$$\begin{aligned} \bar{g}(y)_{i,j} &= \frac{\partial}{\partial y^j}\int_0^1 g((1-\xi)x+\xi y)_i \,\mathrm{d}\xi = \int_0^1 \frac{\partial}{\partial y^j} g((1-\xi)x+\xi y)_i \,\mathrm{d}\xi \\ &= \int_0^1 \xi g((1-\xi)x+\xi y)_{i,j} \,\mathrm{d}\xi = \int_0^1 \xi g((1-\xi)x+\xi y)_{j,i} \,\mathrm{d}\xi \\ &= \bar{g}(y)_{j,i}. \end{aligned}$$

□

Proposition 5.1. *The AVF discrete gradient is the unique discrete gradient satisfying* $(D_2\overline{\nabla}H(x,y))^T = D_2\overline{\nabla}H(x,y)$ *for all H, x and y, and it has the formal expansion*

$$\overline{\nabla}_{AVF}H(x,y) = \sum_{\kappa=0}^{\infty}\frac{1}{(\kappa+1)!}D^{\kappa}\nabla H(x)(y-x)^{\kappa}. \tag{5.27}$$

Proof. Assume that ∇H is an analytic function. As in the proof of Lemma 5.2, let $I = \{i_1, i_2, \dots, i_\kappa\}$ be an ordered set of κ indices, and let I_m be I with the m^{th} element excluded. If $\bar{g}(y)_{i,j} = \bar{g}(y)_{j,i}$ for all i, j, then also

$$\bar{g}(y)_{i,I} = \bar{g}(y)_{i_m,\{i,I_m\}} \quad \text{for all } i, I, m. \tag{5.28}$$

Inserting (5.28) in (5.22) we get $g^{(\kappa)}(x) = (1+\kappa)\bar{g}^{(\kappa)}(x)$. Then inserting this for $\bar{g}^{(\kappa)}(x)$ in (5.14), we get (5.27), which uniquely defines the AVF discrete gradient. □

A consequence of the above result is that the AVF discrete gradient is the unique discrete gradient for which the scheme (5.9) with $\overline{S}(x,\hat{x},h) = S$ is a B-series method when applied to the system (5.6). Furthermore, from the Integrability Lemma (see e.g. [13, Lemma VI.2.7]) and the above, we have that it is the only discrete gradient which defines a gradient vector field in general:

Corollary 1. The AVF discrete gradient is the gradient with respect to the second argument of a function $\tilde{H}(x,y)$. That is,

$$\overline{\nabla}_{\text{AVF}}H(x,y) = \nabla_2\tilde{H}(x,y),$$

for some $\tilde{H} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and all $x, y \in \mathbb{R}^d$. The AVF discrete gradient is the unique discrete gradient to have this property for all H.

As we see from the above definitions and discussion, each of the discrete gradients have their advantages and disadvantages. Gonzalez' midpoint discrete gradient is easily calculated from the energy H and the gradient ∇H, but it is in general only once differentiable. The Itoh–Abe discrete gradient does not require knowledge of the gradient, but is only a first order approximation of the gradient. The AVF discrete gradient is the unique discrete gradient whose series expansion is given by the differentials of the gradient. It does however require an integral to be calculated. If that poses a challenge, the SIA or Furihata discrete gradients are second-order alternatives, but the latter is only defined for H of the form (5.25).

5.2.2 Third and fourth order schemes for the constant S case

Consider now only the cases where S is constant, i.e. (5.6). By comparing the Taylor series of the exact solution and that of the discrete gradient method, and by using the properties of the discrete gradient developed above, we may achieve higher order discrete gradient methods.

In search of a third order scheme, we assume that $\hat{x}$ is a third order in h approximation of $x(t_0+h)$, and find

$$\begin{aligned}S\overline{\nabla}H(x,\hat{x}) =& S(\nabla H(x) + D_2\overline{\nabla}H(x,x)(hS\nabla H(x) + \frac{1}{2}h^2 S\nabla^2 H(x)S\nabla H(x) + \mathcal{O}(h^2))\\ &+ \frac{1}{2}D_2^2\overline{\nabla}H(x,x)(hS\nabla H(x) + \mathcal{O}(h^2), hS\nabla H(x) + \mathcal{O}(h^2)) + \mathcal{O}(h^3)\\ =& f + hSD_2\overline{\nabla}Hf + \frac{1}{2}h^2 SD_2\overline{\nabla}Hf'f + \frac{1}{2}h^2 SD_2^2\overline{\nabla}H(f,f) + \mathcal{O}(h^3),\end{aligned}$$

where we have suppressed the argument x of f, $D_2\overline{\nabla}H$ and $D_2^2\overline{\nabla}H$ in the last line. Furthermore, we use that

$$Q(x, x+\gamma hf(x)) = Q(x,x) + \gamma hD_2Q(x,x)(f,\cdot) + \mathcal{O}(h^2)$$

and (5.21) to get

$$\begin{aligned}&SQ(x, x+\gamma hf(x))S\overline{\nabla}H(x,\hat{x})\\ &\quad= SQ(x,x)S\overline{\nabla}H(x,\hat{x}) + \gamma hSD_2Q(x,x)(f, S\overline{\nabla}H(x,\hat{x})) + \mathcal{O}(h^2)\\ &\quad= SQ(x,x)S(\nabla H(x) + D_2\overline{\nabla}H(x,x)(hS\nabla H(x) + \mathcal{O}(h^2))\\ &\qquad + \gamma hSD_2Q(x,x)(f, S(\nabla H(x) + \mathcal{O}(h)) + \mathcal{O}(h^2)\\ &\quad= SQ(x,x)f + hSQ(x,x)SD_2\overline{\nabla}H(x,x)f + \gamma hSD_2Q(x,x)(f,f) + \mathcal{O}(h^2)\\ &\quad= \frac{1}{2}f'f - SD_2\overline{\nabla}Hf + \frac{1}{2}hf'SD_2\overline{\nabla}Hf - hSD_2\overline{\nabla}HSD_2\overline{\nabla}Hf\\ &\qquad + \frac{1}{6}\gamma hf''(f,f) - \frac{1}{2}\gamma hSD_2^2\overline{\nabla}H(f,f) + \mathcal{O}(h^2),\end{aligned}$$

where again we suppress the argument x in the last expression. Thus the discrete gradient scheme (5.9) is of order 3 if $\overline{\nabla}H \in C^2(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$ and $\overline{S}(x,\hat{x},h) = \overline{S}(x,h)$ is given by

$$\begin{aligned}\overline{S}(x,h) =& S + hSQ(x, x+\frac{2}{3}hf(x))S\\ &+ h^2 S\big(Q(x,x)SQ(x,x) - \frac{1}{12}\nabla^2 H(x)S\nabla^2 H(x)\big)S.\end{aligned}$$

Finding an approximation of S that guarantees higher order of the discrete gradient method quickly becomes significantly more complicated, and results in increasingly complicated expressions for $\overline{S}(x,\hat{x},h)$. For example, it can be shown that one fourth order scheme of the form (5.9) is given by any $\overline{\nabla}H \in$

$C^3(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$ and

$$\begin{aligned}\overline{S}(x,h) = S &+ hS\big(\frac{8}{9}Q(x,z_3) + \frac{1}{9}Q(x,x)\big)S \\ &+ h^2 S\big(Q(x,z_2)SQ(x,z_2) - \frac{1}{12}\nabla^2 H(z_1)S\nabla^2 H(z_1)\big)S \\ &+ h^3 S\big(Q(x,x)SQ(x,x)SQ(x,x) \\ &- \frac{1}{12}\nabla^2 H(x)S\nabla^2 H(x)SQ(x,x) \\ &- \frac{1}{12}Q(x,x)S\nabla^2 H(x)S\nabla^2 H(x)\big)S,\end{aligned} \tag{5.29}$$

where

$$z_1 = x + \frac{1}{2}hf(x), \qquad z_2 = x + \frac{2}{3}hf(x), \qquad z_3 = x + \frac{3}{4}hf(z_1).$$

Note that if we choose a symmetric discrete gradient, we have by Lemma 5.1 that $Q(x,x) = 0$, and many of the terms in (5.29) disappear. If we use the AVF discrete gradient, (5.29) simplifies to

$$\overline{S}(x,h) = S - \frac{1}{12}h^2 S\nabla^2 H(z_1)S\nabla^2 H(z_1)S. \tag{5.30}$$

This is very similar to the higher order AVF methods of Quispel and McLaren, as given in [26], applied to (5.4) with S constant: if we replace z_1 in (5.30) by x, we get their third order scheme; if we replace z_1 by $\frac{x+\hat{x}}{2}$, we get their symmetric fourth order scheme.

Seeing as (5.29) simplifies considerably when the AVF discrete gradient is chosen, and since we in this case get a B-series method, we begin our generalization to arbitrary order by studying this case specifically in the chapter to follow.

5.3 A generalization of the AVF method

Let us recall the concept of B-series. Referring to the definitions in [13, Section III.1], we let T be the set of rooted trees, built recursively from starting with $\tau = \bullet$ and letting $\tau = [\tau_1, \ldots, \tau_m]$ be the tree obtained by grafting the roots of the trees $\tau_1, \ldots, \tau_m$ to a new root. Furthermore, $F(\tau)$ is the elementary differential associated with the tree τ, defined by $F(\bullet)(x) = f(x)$ and

$$F(\tau)(x) = f^{(m)}(x)\big(F(\tau_1)(x), \ldots, F(\tau_m)(x)\big),$$

and $\sigma(\tau)$ is the symmetry coefficient for τ, defined by $\sigma(\bullet) = 1$ and

$$\sigma(\tau) = \sigma(\tau_1)\cdots\sigma(\tau_m)\cdot\mu_1!\mu_2!\cdots, \tag{5.31}$$

where the integers μ_1, $\mu_2, \ldots$ count equal trees among $\tau_1, \ldots, \tau_m$. Then, if $\phi\colon T \cup \{\emptyset\} \to \mathbb{R}$ is an arbitrary map, a *B-series* is a formal series

$$B(\phi, x) = \phi(\emptyset)x + \sum_{\tau \in T} \frac{h^{|\tau|}}{\sigma(\tau)} \phi(\tau) F(\tau)(x). \tag{5.32}$$

The exact solution of (5.1) can be written as the B-series $B(\frac{1}{\gamma}, x)$, where the coefficient γ satisfies $\gamma(\emptyset) = \gamma(\bullet) = 1$ and

$$\gamma(\tau) = |\tau| \gamma(\tau_1) \cdots \gamma(\tau_m) \qquad \text{for } \tau = [\tau_1, \ldots, \tau_m], \tag{5.33}$$

where $|\tau|$ is the order, i.e. the number of nodes, of τ.

Definition 5.3. The *generalized AVF method* is given by

$$\begin{aligned} \frac{\hat{x} - x}{h} = \Bigg(I + \sum_{n=2}^{p-1} h^n \sum_j b_{nj} \bigg(\prod_{k=1}^{n} f'(z_{njk}) + (-1)^n \prod_{k=1}^{n} f'(z_{nj(n-k+1)}) \bigg) \Bigg) \\ \cdot \int_0^1 f((1-\xi)x + \xi \hat{x}) \, \mathrm{d}\xi, \end{aligned} \tag{5.34}$$

where each $z_{njk} := z_{njk}(x, \hat{x}, h) = B(\phi_{njk}, x)$ can be written as a B-series with $\phi(\emptyset) = 1$.

Note that we may alternatively write (5.34) in the slightly more compact form

$$\begin{aligned} \frac{\hat{x} - x}{h} = \sum_{n=0}^{p-1} h^n \sum_j b_{nj} \left(\prod_{k=1}^{n} f'(z_{njk}) + (-1)^n \prod_{k=1}^{n} f'(z_{nj(n-k+1)}) \right) \\ \cdot \int_0^1 f((1-\xi)x + \xi \hat{x}) \, \mathrm{d}\xi \end{aligned}$$

with $\sum_j b_{0j} = \frac{1}{2}$.

Theorem 5.1. *When applied to* (5.1) *with $f(x) = S\nabla H(x)$, where S is a constant skew-symmetric matrix, the scheme* (5.34) *preserves H, in that $H(\hat{x}) = H(x)$.*

Proof. With $f(x) = S\nabla H(x)$, (5.34) becomes

$$\frac{\hat{x} - x}{h} = \overline{S}(x, \hat{x}, h) \overline{\nabla}_{\mathrm{AVF}} H(x, \hat{x}),$$

with

$$\overline{S}(x, \hat{x}, h) = S + \sum_{n=2}^{p-1} h^n \sum_j b_{nj} \left(\prod_{k=1}^{n} S\nabla^2 H(z_{njk}) + (-1)^n \prod_{k=1}^{n} S\nabla^2 H(z_{nj(n-k+1)}) \right) S.$$

We have

$$\begin{aligned}
&\left(\prod_{k=1}^{n} S\nabla^2 H(z_{njk})\cdot S + (-1)^n \prod_{k=1}^{n} S\nabla^2 H(z_{nj(n-k+1)})\cdot S\right)^T \\
&\qquad = S^T \prod_{k=1}^{n} \left(\nabla^2 H(z_{nj(n-k+1)})^T S^T\right) + (-1)^n S^T \prod_{k=1}^{n} \left(\nabla^2 H(z_{njk})^T S^T\right) \\
&\qquad = (-1)^{i+1} S \prod_{k=1}^{n} \nabla^2 H(z_{nj(n-k+1)}) S - S \prod_{k=1}^{n} \nabla^2 H(z_{njk}) S \\
&\qquad = -\left(\prod_{k=1}^{n} S\nabla^2 H(z_{njk})\cdot S + (-1)^n \prod_{k=1}^{n} S\nabla^2 H(z_{nj(n-k+1)})\cdot S\right),
\end{aligned}$$

and thus $\overline{S}(x,\hat{x},h)$ is a skew-symmetric matrix. □

Before considering the order conditions of the generalized AVF method, let us recall a couple of results from the literature on B-series.

Lemma 5.4 ([13, Lemma III.1.9]). *Let $B(a,x)$ be a B-series with $a(\emptyset)=1$. Then $hf(B(a,x)) = B(a',x)$ is also a B-series, with $a'(\emptyset)=0$, $a'(\bullet)=1$ and otherwise*

$$a'(\tau) = a(\tau_1)\cdots a(\tau_m) \quad \textit{for } \tau = [\tau_1,\ldots,\tau_m].$$

Lemma 5.5 ([23, Theorem 2.2]). *Let $B(a,x)$ and $B(b,x)$ be two B-series with $a(\emptyset)=1$ and $b(\emptyset)=0$. Then $hf'(B(a,x))B(b,x) = B(a\times b,x)$, i.e. a B-series, with $(a\times b)(\emptyset) = (a\times b)(\bullet) = 0$ and otherwise*

$$(a\times b)(\tau) = \sum_{i=1}^{m} \prod_{j=1, j\neq i}^{m} a(\tau_j) b(\tau_i) \quad \textit{for } \tau = [\tau_1,\ldots,\tau_m].$$

Proposition 1 in [2] states that the standard AVF method is a B-series method. We build on the proof of that proposition to prove the following result.

Proposition 5.2. *The generalized AVF method* (5.34) *is a B-series method.*

Proof. First we define $\hat{e}: T\cup\{\emptyset\}\to\mathbb{R}$ by $\hat{e}(\emptyset)=1$ and $\hat{e}(\tau)=0$ for all $\tau\neq\emptyset$. Then, assuming that the solution $\hat{x}$ of (5.34) can be written as the B-series $\hat{x} = B(\Phi,x)$, we find the B-series

$$\begin{aligned}
h\int_0^1 f\big((1-\xi)x+\xi\hat{x}\big)\,\mathrm{d}\xi &= h\int_0^1 f\big(B((1-\xi)\hat{e}+\xi\Phi,x)\big)\,\mathrm{d}\xi \\
&= \int_0^1 B\big(((1-\xi)\hat{e}+\xi\Phi)',x\big)\,\mathrm{d}\xi \\
&= B\Big(\int_0^1 ((1-\xi)\hat{e}+\xi\Phi)'\,\mathrm{d}\xi, x\Big).
\end{aligned}$$

Setting $\theta := \int_0^1 ((1-\xi)\hat{e}+\xi\Phi)'\,\mathrm{d}\xi = \int_0^1 ((1-\xi)\hat{e})'\,\mathrm{d}\xi + \int_0^1 (\xi\Phi)'\,\mathrm{d}\xi = \int_0^1 (\xi\Phi)'\,\mathrm{d}\xi$, we get

$$\theta(\emptyset)=0, \quad \theta(\bullet)=1, \quad \theta([\tau_1,\dots,\tau_m]) = \frac{1}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m). \tag{5.35}$$

Then we may rewrite (5.34) as

$$\begin{aligned}
\hat{x} &= x + \Big(I + \sum_{n=2}^{p-1} h^n \sum_j b_{nj}\big(\prod_{k=1}^{n} f'(B(\phi_{njk},x)) \\
&\quad + (-1)^n \prod_{k=1}^{n} f'(B(\phi_{nj(n-k+1)},x))\big)\Big) B(\theta,x) \\
&= x + B(\theta,x) + \sum_{n=2}^{p-1}\sum_j b_{nj}\big(B(\phi_{nj1}\times\cdots\times\phi_{njn}\times\theta,x) \\
&\quad + (-1)^n B(\phi_{njn}\times\cdots\times\phi_{nj1}\times\theta,x)\big) \\
&= B(\Phi,x),
\end{aligned}$$

with

$$\begin{aligned}
\Phi = \hat{e} + \theta + \sum_{n=2}^{p-1}\sum_j b_{nj}\big(&\phi_{nj1}\times\cdots\times\phi_{njn}\times\theta \\
&+ (-1)^n \phi_{njn}\times\cdots\times\phi_{nj1}\times\theta\big).
\end{aligned} \tag{5.36}$$

□

Comparing the B-series of the exact solution and the B-series of the solution of (5.34), and noting that the elementary differentials are independent, we immediately get the following result.

Theorem 5.2. *The generalized AVF method* (5.34) *is of order p if and only if*

$$\Phi(\tau) = \frac{1}{\gamma(\tau)} \qquad \textit{for } |\tau| \le p, \tag{5.37}$$

where Φ is given by (5.36) *and γ is given by* (5.33).

The terms $\Phi(\tau)$ can be found from (5.36) by applying Lemma 5.5 recursively, as illustrated by the following example.

Example 5.1. *Consider $\tau = [\bullet,[[\bullet]]]$, and assume we have found Φ for all trees up to and including order four already, as given in Table 5.1. We have*

$$\theta([\bullet,[[\bullet]]]) = \frac{1}{3}\Phi(\bullet)\Phi([[\bullet]]) = \frac{1}{3}(\frac{1}{4}+2\sum_j b_{2j}) = \frac{1}{12}+\frac{2}{3}\sum_j b_{2j}.$$

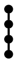
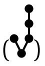
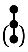
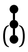
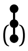
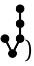
$\lvert\tau\rvert$	$F(\tau)^i$	τ	$\sigma(\tau)$	$\gamma(\tau)$	$\Phi(\tau)$
1	f^i	$\bullet$	1	1	1
2	$f^i_j f^j$		1	2	$\frac{1}{2}$
3	$f^i_{jk} f^j f^k$		2	3	$\frac{1}{3}$
	$f^i_j f^j_k f^k$		1	6	$\frac{1}{4}+2\sum_j b_{2j}$
4	$f^i_{jkl} f^j f^k f^l$		6	4	$\frac{1}{4}$
	$f^i_{jk} f^j f^k_l f^l$		1	8	$\frac{1}{6}+\sum_{j,k} b_{2j}\phi_{2jk}(\bullet)$
	$f^i_j f^j_{kl} f^k f^l$		2	12	$\frac{1}{6}+2\sum_{j,k} b_{2j}\phi_{2j1}(\bullet)$
	$f^i_j f^j_k f^k_l f^l$		1	24	$\frac{1}{8}+2\sum_j b_{2j}$

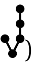
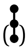
Table 5.1: Elementary differentials and their coefficients in the B-series of the solution of (5.34), up to fourth order.

Then we calculate

$(\phi_{2j1}\times\phi_{2j2}\times\theta)($  $) = \phi_{2j1}(\bullet)(\phi_{2j2}\times\theta)($  $) + \phi_{2j1}($  $)(\phi_{2j2}\times\theta)(\bullet)$

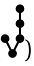
$= \phi_{2j1}(\bullet)(\phi_{2j2}\times\theta)($  $) = \phi_{2j1}(\bullet)\phi_{2j2}(\emptyset)\theta($ $) = \frac{1}{2}\phi_{2j1}(\bullet),$

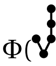
where we have used in the second equality that $(\phi_{2j2}\times\theta)(\bullet)=\phi_{2j2}(\emptyset)\theta(\emptyset)=0$. *Similarly we find* $(\phi_{2j2}\times\phi_{2j1}\times\theta)($  $)=\frac{1}{2}\phi_{2j2}(\bullet)$. *Furthermore,*

$(\phi_{3j1}\times\phi_{3j2}\times\phi_{3j3}\times\theta)($  $) = \phi_{3j1}(\bullet)(\phi_{3j2}\times\phi_{3j3}\times\theta)($  $)$

$= \phi_{3j1}(\bullet)\phi_{3j2}(\emptyset)(\phi_{3j3}\times\theta)($ $)$

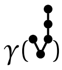
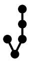
$$= \phi_{3j1}(\bullet)\phi_{3j3}(\emptyset)\theta(\bullet) = \phi_{3j1}(\bullet),$$

and $(\phi_{3j3}\times\phi_{3j2}\times\phi_{3j1}\times\theta)($  $)=\phi_{3j3}(\bullet)$. *Hence,*

$\Phi($  $) = \frac{1}{12}+\frac{2}{3}\sum_j b_{2j}+\frac{1}{2}\sum_j b_{2j}(\phi_{2j1}(\bullet)+\phi_{2j2}(\bullet))+\sum_j b_{3j}(\phi_{3j1}(\bullet)-\phi_{3j3}(\bullet)).$

Now, if we assume the order condition (5.37) *to be satisfied for all trees up to and including order four, we can replace*

$$\sum_j b_{2j} = -\frac{1}{24} \quad \textit{and} \quad \sum_j b_{2j}(\phi_{2j1}(\bullet)+\phi_{2j2}(\bullet)) = -\frac{1}{24}$$

in the above expression, use that $\gamma($$) = 30$, *and get that* (5.37) *is satisfied for*  *if and only if*

$$\sum_j b_{3j}(\phi_{3j1}(\bullet) - \phi_{3j3}(\bullet)) = -\frac{1}{720}. \tag{5.38}$$

5.3.1 Construction of higher order schemes

As the size of the trees grows, finding $\Phi(\tau)$ from (5.36) can become quite a cumbersome operation. Furthermore, we observe from Table 5.1 that there are some equivalent order conditions for different trees. Before presenting more convenient techniques for finding order conditions for the generalized AVF method, let us define some more concepts related to B-series and trees.

First, recall that the Butcher product of two trees $u = [u_1,\dots,u_m]$ and $v = [v_1,\dots,v_n]$ is given by $u \circ v = [u_1,u_2,\dots,u_m,v]$. This operation is neither associative nor commutative, and in contrast to the practice in [13], we here take the product of several factors without parentheses to mean evaluation from right to left:

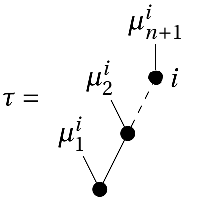
$$u_1 \circ u_2 \circ \cdots \circ u_k := u \circ (u_2 \circ (\cdots \circ u_k)).$$

Given a forest $\mu = (\tau_1,\dots,\tau_m)$, the tree obtained by grafting the roots of every tree in μ to a new root is denoted by $[\mu] = [\tau_1,\dots,\tau_m]$. Moreover, $\mu^{-1}(\tau)$ denotes the forest such that $[\mu^{-1}(\tau)] = \tau$. We extend the maps $\phi : T \cup \{\emptyset\} \to \mathbb{R}$ and $\gamma : T \cup \{\emptyset\} \to \mathbb{R}$ to forests by the letting $\phi(\mu) = \prod_{i=1}^m \phi(\tau_i)$ and $\gamma(\mu) = \prod_{i=1}^m \gamma(\tau_i)$ for $\mu = (\tau_1,\dots,\tau_m)$.

Consider now a tree τ consisting of $|\tau|$ nodes. We may number every tree from 1 to $|\tau|$, starting at the root and going from left to right on the increasing levels above. For a given node $i \in [1,\dots,|\tau|]$ on level $n+1$, there exists a unique set of forests $\hat{\tau}^i = \{\mu_1^i,\dots,\mu_{n+1}^i\}$ such that

$$\tau = [\mu_1^i] \circ [\mu_2^i] \circ \cdots \circ [\mu_{n+1}^i].$$

That is, labeling node i,



Proposition 5.3. *The* Φ *of* (5.37) *can alternatively be found by*

$$\Phi(\tau) = \hat{e}(\tau) + \theta(\tau) + \sum_{i \text{ s.t. } n \geq 2} \Lambda(\hat{\tau}^i) \tag{5.39}$$

where $\hat{e}(\emptyset)=1$ *and* $\hat{e}(\tau)=0$ *for all* $\tau \neq \emptyset$, $\theta(\emptyset)=0$, $\theta(\bullet)=1$,

$$\theta([\tau_1,\dots,\tau_m]) = \frac{1}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m),$$

and

$$\begin{aligned}\Lambda(\hat{\tau}^i) = {} & \theta([\mu^i_{n+1}])\sum_j b_{nj}\big(\phi_{nj1}(\mu^i_1)\cdots\phi_{njn}(\mu^i_n) \\ & + (-1)^n\phi_{njn}(\mu^i_1)\cdots\phi_{nj1}(\mu^i_n)\big).\end{aligned} \tag{5.40}$$

Proof. Define n_i so that n_i+1 is the level of node i. Collect the children of node i in the set C_i. We have

$$[\mu^i_{n_i+1}] = [\mu^k_{n_k}]\circ[\mu^k_{n_k+1}] \qquad \text{for all } k \in C_i,$$

and thus

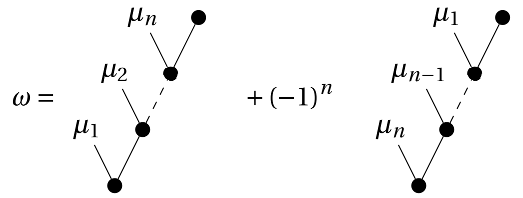
$$(a\times b)([\mu^i_{n_i+1}]) = \sum_{k\in C_i} a(\mu^k_{n_k})b([\mu^k_{n_k+1}]).$$

Note also that $\mu^i_{n_i} = \mu^k_{n_i} = \mu^k_{n_k-1}$ if $k \in C_i$. Then we get

$$\begin{aligned}(\phi_{nj1}\times\cdots\times\phi_{njn}\times\theta)(\tau) &= (\phi_{nj1}\times\cdots\times\phi_{njn}\times\theta)([\mu^1_1]) \\ &= \sum_{i_1\in C_1}\phi_{nj1}(\mu^{i_1}_1)(\phi_{nj2}\times\cdots\times\phi_{njn}\times\theta)([\mu^{i_1}_2]) \\ &= \sum_{i_1\in C_1}\phi_{nj1}(\mu^{i_1}_1)\sum_{i_2\in C_{i_1}}\phi_{nj2}(\mu^{i_2}_2)(\phi_{nj3}\times\cdots\times\phi_{njn}\times\theta)([\mu^{i_2}_3]) \\ &= \sum_{i_1\in C_1}\sum_{i_2\in C_{i_1}}\phi_{nj1}(\mu^{i_2}_1)\phi_{nj2}(\mu^{i_2}_2)(\phi_{nj3}\times\cdots\times\phi_{njn}\times\theta)([\mu^{i_2}_3]) \\ &\;\;\vdots \\ &= \sum_{i_1\in C_1}\sum_{i_2\in C_{i_1}}\cdots\sum_{i_n\in C_{i_{n-1}}}\phi_{nj1}(\mu^{i_n}_1)\cdots\phi_{njn}(\mu^{i_n}_n)\theta([\mu^{i_n}_{n+1}]) \\ &= \sum_{i \text{ on level } n+1}\phi_{nj1}(\mu^i_1)\cdots\phi_{njn}(\mu^i_n)\theta([\mu^i_{n+1}]).\end{aligned}$$

Inserting this and the corresponding result for $(\phi_{njn}\times\cdots\times\phi_{nj1}\times\theta)(\tau)$ in (5.36), we get (5.40). □

In [3, 8], conditions are derived for a B-series method to be energy preserving when applied to the system (5.6). In [26], while giving the AVF method as one such method, Quispel and McLaren present a general form of what they call energy-preserving linear combinations of rooted trees:



Here we give their result as a lemma, which is proved later by the proof of the more general Theorem 5.5.

Lemma 5.6. *Let $\mu_1, \dots, \mu_n$ be n arbitrary forests. Then, if $f(x) = S\nabla H(x)$ for some skew-symmetric constant matrix S, we have that $F(\omega)(x) \cdot \nabla H(x) = 0$ for*

$$\omega = [\mu_1] \circ [\mu_2] \circ \cdots [\mu_n] \circ [\emptyset] + (-1)^n [\mu_n] \circ [\mu_{n-1}] \circ \cdots [\mu_1] \circ [\emptyset]. \qquad (5.41)$$

There is a connection between (5.39) and Lemma 5.6 such that instead of order conditions for every tree, we can calculate order conditions for every energy-preserving linear combination. To see this we start by collecting the leaf nodes, i.e. nodes with no children, of the tree τ in a set I_l and the other nodes in the set I_n. If node $i \in I_n$, we may then use the relation

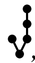
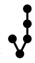
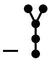
$$\Lambda(\{\mu_1^i, \dots, \mu_n^i, \mu_{n+1}^i\}) = \theta([\mu_{n+1}^i])\Lambda(\{\mu_1^i, \dots, \mu_n^i, \emptyset\})$$

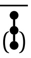
to find $\Lambda(\hat{\tau}^i)$ from the previously calculated Λ for a smaller tree. Then if lower order conditions are satisfied, we have numerical values for these Λ. The leaf nodes on the other hand, with their corresponding $\hat{\tau}^i = \{\mu_1^i, \dots, \mu_n^i, \emptyset\}$, gives an energy-preserving linear combination (5.41) which τ belongs to. If i is on level two, this combination is simply $\tau - \tau = 0$, and accordingly Λ is not calculated for these nodes in (5.39). Moreover, leaves on the same level have identical $\hat{\tau}^i$. Thus, a tree with leaves on m different levels above level two will belong to at most m non-zero energy-preserving linear combinations (5.41).

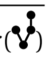
If we assume the conditions for order $< p$ to be satisfied, we may replace (5.37) by

$$\sum_{i \in I_l} \Lambda(\hat{\tau}^i) = \frac{1}{\gamma(\tau)} - \hat{e}(\tau) - \sum_{i \in I_n} \frac{\Lambda(\{\mu_1^i, \dots, \mu_n^i, \emptyset\})}{(|\mu_{n+1}^i| + 1)\gamma(\mu_{n+1}^i)}, \qquad (5.42)$$

where $|\mu|$ denotes the number of trees in the forest μ. Note that $\Lambda(\{\emptyset\}) = 1$ and hence $\Lambda(\hat{\tau}^1) = \theta(\tau)$. Then we can calculate the numerical value for the right hand side and, if τ has leaves on only one level > 2, find an order condition for both τ and the other tree in the combination (5.41). This warrants an example.

Example 5.2. *Consider again the tree $\tau =$* *, which is part of the energy-preserving linear combination $\omega =$*  $-$ *. Ignoring the two nodes on level 2, there are three nodes to calculate Λ for: $i = 1$, $i = 4$ and $i = 5$. We find*

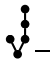
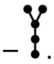
$\Lambda(\hat{\tau}^1) = 1 \,/\, \big((2+1)\gamma(\bullet)\gamma($  $)\big) = \frac{1}{3 \cdot 1 \cdot 6} = \frac{1}{18}$

$\Lambda(\hat{\tau}^4) = \frac{1}{(1+1)\gamma(\bullet)}\Lambda(\{\bullet, \emptyset, \emptyset\}) = \frac{1}{2\gamma(\bullet)}\Big(1 \,/\, \gamma($  $) - 1 \,/\, \big(3\gamma(\bullet)\gamma($ $)\big)\Big) = \frac{1}{2}\left(\frac{1}{8} - \frac{1}{6}\right) = -\frac{1}{48},$

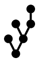
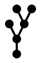
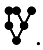
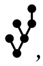
$$\Lambda(\hat{\tau}^5) = \Lambda(\{\bullet, \emptyset, \emptyset, \emptyset\}) = \sum_j b_{3j}\big(\phi_{3j1}(\bullet) - \phi_{3j3}(\bullet)\big).$$

The right hand side of (5.42) *becomes*

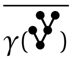
$$\frac{1}{\gamma(\tau)}-\frac{1}{18}-(-\frac{1}{48})=\frac{1}{30}-\frac{1}{18}+\frac{1}{48}=-\frac{1}{720},$$

and we have the order condition (5.38) *for the linear combination*  – .

If there are leaves on $r>1$ different levels levels above level two, things get slightly more complicated. Then we get r different terms on the left hand side of (5.42) and we need to consider the order condition for τ and the r trees it forms energy-preserving linear combinations with, so that we get an equation for every energy-preserving combination of these trees, also those not including τ. This is illustrated by the following example.

Example 5.3. *The tree*  *forms energy-preserving combinations with both*  *and* *. Thus we have to calculate* (5.42) *for all three trees to find order conditions for the corresponding linear combinations. Starting with* $\tau=$ *, which has three nodes above level two, two leaves and one non-leaf, we get*

$\Lambda(\hat{\tau}^4)=\Lambda(\{\bullet,$ $,\emptyset\})=\sum_j b_{2j}\big(\phi_{2j1}(\bullet)\phi_{2j2}($ $)+\phi_{2j2}(\bullet)\phi_{2j1}($ $)\big),$

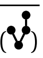
$\Lambda(\hat{\tau}^5)=\frac{1}{(1+1)\gamma(\bullet)}\Lambda(\{\bullet,\bullet,\emptyset\})=\frac{1}{2\gamma(\bullet)}\frac{1}{2}\Bigg(\frac{1}{\gamma(}$  $\frac{)}{}-\frac{1}{3\gamma(\bullet)\gamma(}$ $\frac{)}{}\Bigg)$

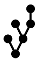
$$=\frac{1}{2}\frac{1}{2}\Big(\frac{1}{15}-\frac{1}{9}\Big)=-\frac{1}{90},$$

$$\Lambda(\hat{\tau}^6)=\Lambda(\{\bullet,\bullet,\emptyset,\emptyset\})=\sum_j b_{3j}\big(\phi_{3j1}(\bullet)\phi_{3j2}(\bullet)-\phi_{3j3}(\bullet)\phi_{3j2}(\bullet)\big)$$

$$=\sum_j b_{3j}\phi_{3j2}(\bullet)(\phi_{3j1}-\phi_{3j3})(\bullet).$$

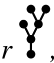
For the right hand side of (5.42)*, we get*

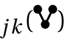
$\frac{1}{\gamma(\tau)}-\frac{1}{(2+1)\gamma(\bullet)\gamma(}$  $\frac{)}{}-(-\frac{1}{90})=\frac{1}{48}-\frac{1}{3\cdot 1\cdot 8}+\frac{1}{90}=-\frac{7}{720},$

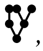
and hence the order condition for  *is*

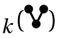
$\sum_j b_{2j}\big(\phi_{2j1}(\bullet)\phi_{2j2}($ $)+\phi_{2j2}(\bullet)\phi_{2j1}($ $)\big)$

$$+\sum_j b_{3j}\phi_{3j2}(\bullet)(\phi_{3j1}-\phi_{3j3})(\bullet)=-\frac{7}{720}. \tag{5.43}$$

Similarly we calculate (5.42) *for*  ,

$\sum_{jk} b_{2j}\phi_{2jk}($$) - 2\sum_{j} b_{3j}\phi_{3j2}(\bullet)(\phi_{3j1} - \phi_{3j3})(\bullet) = -\frac{1}{120},$ (5.44)

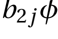
and for  ,

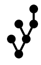
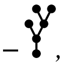
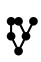
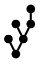
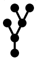
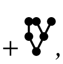
$\sum_{jk} b_{2j}\phi_{2jk}($$) + 2\sum_{j} b_{2j}\big(\phi_{2j1}(\bullet)\phi_{2j2}($ $) + \phi_{2j2}(\bullet)\phi_{2j1}($ $)\big) = -\frac{1}{36}.$ (5.45)

Combining (5.43), (5.44) *and* (5.45), *we get the equivalent system of equations*

$$\sum_{j} b_{3j}\phi_{3j2}(\bullet)(\phi_{3j1}(\bullet) - \phi_{3j3}(\bullet)) = \frac{1}{240} + \alpha, \tag{5.46}$$

$\sum_{j} b_{2j}(\phi_{2j1}(\bullet)\phi_{2j2}($ $) + \phi_{2j1}($ $)\phi_{2j2}(\bullet)) = -\frac{1}{72} - \alpha,$ (5.47)

$\sum_{j,k} b_{2j}\phi_{2jk}($$) = 2\alpha,$ (5.48)

where the choice of α ∈ ℝ is arbitrary. The order conditions (5.46)–(5.48) *can be associated to the linear combinations*  –  ,  +  *and*  +  , *respectively.*

By considering the order conditions in Table 5.2, we find a fifth order scheme of the form (5.34) given by

$$\begin{aligned}\frac{\hat{x}-x}{h} = \Big(I &- \frac{5}{136}h^2\big(f'(z_2)f'(z_3) + f'(z_3)f'(z_2)\big) - \frac{1}{102}h^2 f'(x)f'(x) \\ &+ \frac{1}{288}h^3\big(f'(x)f'(x)f'(z_1) + f'(z_1)f'(x)f'(x)\big) \\ &+ \frac{1}{120}h^4 f'(x)f'(x)f'(x)f'(x)\Big)\int_0^1 f((1-\xi)x + \xi\hat{x})\,\mathrm{d}\xi,\end{aligned} \tag{5.49}$$

where

$$z_1 = x + \frac{2}{5}hf(x), \qquad z_2 = x + \frac{17+\sqrt{17}}{30}hf(z_1), \qquad z_3 = x + \frac{17-\sqrt{17}}{30}hf(z_1).$$

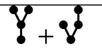
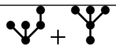
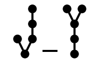
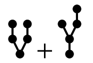
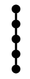
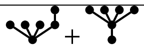
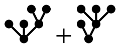
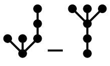
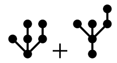
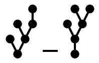
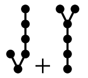
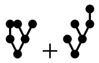
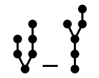
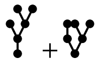
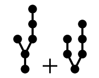
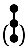
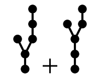
$\lvert\tau\rvert$	ω	Order condition
1	•	–
2	–	–
3		$\sum_j b_{2j} = -\frac{1}{24}$
4		$\sum_{j,k} b_{2j}\phi_{2jk}(\bullet) = -\frac{1}{24}$
5		$\sum_{j,k} b_{2j}\phi_{2jk}(\bullet)^2 = -\frac{1}{40}$
		$\sum_j b_{2j}\phi_{2j1}(\bullet)\phi_{2j2}(\bullet) = -\frac{1}{90}$
		$\sum_j b_{3j}(\phi_{3j1}(\bullet) - \phi_{3j3}(\bullet)) = -\frac{1}{720}$
		$\sum_{j,k} b_{2j}\phi_{2jk}($ $) = -\frac{1}{60}$
		$\sum_j b_{4j} = \frac{1}{240}$
6		$\sum_{j,k} b_{2j}\phi_{2jk}(\bullet)^3 = -\frac{1}{60}$
		$\sum_j b_{2j}(\phi_{2j1}(\bullet)^2\phi_{2j2}(\bullet) + \phi_{2j1}(\bullet)\phi_{2j2}(\bullet)^2) = -\frac{1}{72}$
		$\sum_j b_{3j}(\phi_{3j1}(\bullet)^2 - \phi_{3j3}(\bullet)^2) = -\frac{1}{720}$
		$\sum_{j,k} b_{2j}\phi_{2jk}(\bullet)\phi_{2jk}($ $) = -\frac{1}{96}$
		$\sum_j b_{3j}\phi_{3j2}(\bullet)(\phi_{3j1}(\bullet) - \phi_{3j3}(\bullet)) = \frac{1}{240} + \alpha_1$
		$\sum_j b_{4j}(\phi_{4j1}(\bullet) + \phi_{4j4}(\bullet)) = \frac{1}{240}$
		$\sum_j b_{2j}(\phi_{2j1}(\bullet)\phi_{2j2}($ $) + \phi_{2j1}($ $)\phi_{2j2}(\bullet)) = -\frac{1}{72} - \alpha_1$
		$\sum_j b_{3j}(\phi_{3j1}($ $) - \phi_{3j3}($ $)) = -\frac{1}{180} - \alpha_2$
		$\sum_{j,k} b_{2j}\phi_{2jk}($ $) = 2\alpha_1$
		$\sum_{j,k} b_{2j}\phi_{2jk}($$) = \alpha_2$
		$\sum_j b_{4j}(\phi_{4j2}(\bullet) + \phi_{4j3}(\bullet)) = -\frac{1}{1440} - \alpha_2$

Table 5.2: Energy-preserving linear combinations of elementary differentials, and their associated order conditions for the scheme (5.34), up to sixth order. The coefficients $\alpha_1, \alpha_2 \in \mathbb{R}$ are arbitrary.

A symmetric sixth order scheme is given by

$$
\begin{aligned}
\frac{\hat{x}-x}{h} = \Big(I &- \frac{13}{360}h^2 f'\big(\bar{x}+\frac{\sqrt{13}}{26}hf(\bar{x}-\frac{3\sqrt{13}}{26}hf(\bar{x}))\big) \\
&\cdot f'\big(\bar{x}-\frac{\sqrt{13}}{26}hf(\bar{x}+\frac{3\sqrt{13}}{26}hf(\bar{x}))\big) \\
&- \frac{13}{360}h^2 f'\big(\bar{x}-\frac{\sqrt{13}}{26}hf(\bar{x}+\frac{3\sqrt{13}}{26}hf(\bar{x}))\big) \\
&\cdot f'\big(\bar{x}+\frac{\sqrt{13}}{26}hf(\bar{x}-\frac{3\sqrt{13}}{26}hf(\bar{x}))\big) \\
&- \frac{1}{180}h^2 f'(x)f'(x) - \frac{1}{180}h^2 f'(\hat{x})f'(\hat{x}) \\
&+ \frac{1}{720}h^3 f'(\bar{x}-\frac{1}{2}hf(\bar{x}))f'(\bar{x})f'(\bar{x}+\frac{1}{2}hf(\bar{x})) \\
&- \frac{1}{720}h^3 f'(\bar{x}+\frac{1}{2}hf(\bar{x}))f'(\bar{x})f'(\bar{x}-\frac{1}{2}hf(\bar{x})) \\
&+ \frac{1}{120}h^4 f'(\bar{x})f'(\bar{x})f'(\bar{x})f'(\bar{x})\Big)\int_0^1 f((1-\xi)x+\xi\hat{x})\,\mathrm{d}\xi,
\end{aligned}
\tag{5.50}
$$

where $\bar{x} = \frac{x+\hat{x}}{2}$. If we wish to calculate the matrix in front of the integral explicitly, we have a non-symmetric sixth order scheme given by

$$
\begin{aligned}
\frac{\hat{x}-x}{h} = \Big(I &- \frac{13}{360}h^2\big(f'(z_6)f'(z_7)+f'(z_7)f'(z_6)\big) \\
&- \frac{1}{180}h^2\big(f'(x)f'(x)+f'(z_1)f'(z_1)\big) \\
&+ \frac{1}{720}h^3\big(f'(x)f'(z_2)f'(z_3)-f'(z_3)f'(z_2)f'(x)\big) \\
&+ \frac{1}{120}h^4 f'(z_2)f'(z_2)f'(z_2)f'(z_2)\Big)\int_0^1 f((1-\xi)x+\xi\hat{x})\,\mathrm{d}\xi,
\end{aligned}
\tag{5.51}
$$

with

$$
\begin{aligned}
z_1 &= x+\frac{1}{4}hf(x)+\frac{3}{4}hf\big(x+\frac{2}{3}hf(x+\frac{1}{3}hf(x))\big), \\
z_2 &= x+\frac{1}{2}hf(x), \qquad z_3 = x+hf(z_2), \\
z_4 &= \frac{1}{2}(x+z_3)-\frac{3\sqrt{13}}{26}hf(z_2), \qquad z_5 = \frac{1}{2}(x+z_3)+\frac{3\sqrt{13}}{26}hf(z_2), \\
z_6 &= \frac{1}{2}(x+z_1)+\frac{\sqrt{13}}{26}hf(z_4), \qquad z_7 = \frac{1}{2}(x+z_1)-\frac{\sqrt{13}}{26}hf(z_5).
\end{aligned}
$$

5.4 AVF discrete gradient methods for general skew-gradient systems

We will now build on the results of the previous paper by generalizing the results to the situation where $S(x)$ in the skew-gradient system (5.4) is not necessarily constant. Consider therefore now an ODE of the form (5.4), and set again $g := \nabla H$. By Taylor expansion of x around $t = t_0$ we get

$$\begin{aligned} x(t_0+h) = {} & x + hSg + \frac{h^2}{2}(S'gSg + Sg'Sg) + \frac{h^3}{6}(S''g(Sg,Sg) + 2S'g'(Sg,Sg) \\ & + Sg''(Sg,Sg) + S'gS'gSg + S'gSg'Sg + Sg'S'gSg + Sg'Sg'Sg) \\ & + \mathcal{O}(h^4), \end{aligned}$$

where $x := x(t_0)$, and S, g and their derivatives are evaluated in x. Introducing the notation $f^\circ := S'g$ and $f^\bullet := Sg'$, we can write this in the abbreviated form

$$\begin{aligned} x(t_0+h) = {} & x + hf + \frac{h^2}{2}(f^\circ f + f^\bullet f) + \frac{h^3}{6}(f^{\circ\circ}(f,f) + 2f^{\circ\bullet}(f,f) \\ & + f^{\bullet\bullet}(f,f) + f^\circ f^\circ f + f^\circ f^\bullet f + f^\bullet f^\circ f + f^\bullet f^\bullet f) + \mathcal{O}(h^4). \end{aligned} \tag{5.52}$$

5.4.1 Skew-gradient systems and P-series

A *P-series* is given by

$$P(\phi,(x,y)) = \begin{pmatrix} \phi(\emptyset)x + \sum_{\tau\in TP_\bullet} \frac{h^{|\tau|}}{\sigma(\tau)}\phi(\tau)F(\tau)(x,y) \\ \phi(\emptyset)y + \sum_{\tau\in TP_\circ} \frac{h^{|\tau|}}{\sigma(\tau)}\phi(\tau)F(\tau)(x,y) \end{pmatrix}, \tag{5.53}$$

where TP is the set of rooted bi-colored trees and $TP_\bullet$ and $TP_\circ$ are the subsets of TP whose roots are black and white, respectively [13, Section III.2]. The bi-colored trees are built recursively; starting with $\bullet$ and $\circ$, we let $\tau = [\tau_1,\ldots,\tau_m]_\bullet$ be the tree you get by grafting the roots of $\tau_1,\ldots,\tau_m$ to a black root and $\tau = [\tau_1,\ldots,\tau_m]_\circ$ the tree you get by grafting $\tau_1,\ldots,\tau_m$ to a white root. No subscript, i.e. $\tau = [\tau_1,\ldots,\tau_m]$, means grafting to a black root.

The exact solution of a partitioned system

$$\begin{aligned} \dot{x} &= f(x,y), \quad x(t_0) = x_0, \\ \dot{y} &= g(x,y), \quad y(t_0) = y_0, \end{aligned} \tag{5.54}$$

can be written as $(x(t_0+h), y(t_0+h)) = P(1/\gamma,(x_0,y_0))$, where the coefficient γ is given by $\gamma(\emptyset) = \gamma(\bullet) = \gamma(\circ) = 1$ and (5.33). As noted in [5], setting $f(x,y) := S(y)\nabla H(x)$, the skew-gradient system (5.4) can be written as (5.54) with $g = f$. When $g = f$, all coefficients and the elementary differentials $F(\tau)$ in (5.53) are given independent of the color of the root. Thus for the system (5.4), it suffices to consider

$$P(\phi, x) = \phi(\emptyset)x + \sum_{\tau\in TP_\bullet} \frac{h^{|\tau|}}{\sigma(\tau)}\phi(\tau)F(\tau)(x), \tag{5.55}$$

and we have that the exact solution of (5.4) can be written as $x(t_0+h) = P(1/\gamma, x_0)$. Breaking slightly with convention, we define a P-series to be the single row version (5.55) in the remainder of this paper. Denoting black-rooted subtrees by τ_i and white-rooted subtrees by $\bar{\tau}_i$, the elementary differentials $F(\tau)$ for the skew-gradient system are given recursively by $F(\bullet)(x) = F(\circ)(x) = S(x)\nabla H(x)$, and

$$F(\tau)(x) = S^{(l)}D^m\nabla H(F(\tau_1)(x),\dots,F(\tau_m)(x),F(\bar{\tau}_1)(x),\dots,F(\bar{\tau}_l)(x)) \tag{5.56}$$

for both $\tau = [\tau_1,\dots,\tau_m,\bar{\tau}_1,\dots,\bar{\tau}_l]_\bullet$ and $\tau = [\tau_1,\dots,\tau_m,\bar{\tau}_1,\dots,\bar{\tau}_l]_\circ$. The bi-colored trees in $TP_\bullet$ and their corresponding elementary differentials F are given up to order three in Table 5.3. The number of trees grows very quickly with the order; see `https://oeis.org/A000151`.

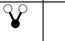
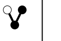
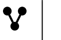
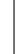
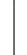
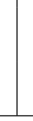
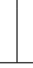
$\vert\tau\vert$	$F(\tau)^i$	$F(\tau)$	τ	$\alpha(\tau)$	$\gamma(\tau)$	$\sigma(\tau)$
1	$S^i_j g^j$	f	•	1	1	1
2	$S^i_{jk} g^j S^k_l g^l$	$f^\circ f$		1	2	1
	$S^i_j g^j_k S^k_l g^l$	$f^\bullet f$		1	2	1
3	$S^i_{jkm} g^j S^k_l g^l S^m_n g^n$	$f^{\circ\circ}(f,f)$		1	3	2
	$S^i_{jk} g^j_m S^k_l g^l S^m_n g^n$	$f^{\circ\bullet}(f,f)$		2	3	1
	$S^i_j g^j_{km} S^k_l g^l S^m_n g^n$	$f^{\bullet\bullet}(f,f)$		1	3	2
	$S^i_{jk} g^j S^k_{lm} g^l S^m_n g^n$	$f^\circ f^\circ f$		1	6	1
	$S^i_j g^j_k S^k_{lm} g^l S^m_n g^n$	$f^\bullet f^\circ f$		1	6	1
	$S^i_{jk} g^j S^k_l g^l_m S^m_n g^n$	$f^\circ f^\bullet f$		1	6	1
	$S^i_j g^j_k S^k_l g^l_m S^m_n g^n$	$f^\bullet f^\bullet f$		1	6	1

Table 5.3: Bi-colored trees and their elementary differentials up to third order.

The following lemma is Lemma III.2.2 in [13] amended to fit our setting.

Lemma 5.7. *Let $P(a,x)$ and $P(b,x)$ be two P-series with $a(\emptyset) = b(\emptyset) = 1$. Then*

$$hS(P(a,x))\nabla H(P(b,x)) = P(a\vee b, x),$$

where $(a\vee b)(\emptyset) = 0$, $(a\vee b)(\bullet) = 1$, and

$$(a\vee b)(\tau) = a(\tau_1)\cdots a(\tau_m)b(\bar{\tau}_1)\cdots b(\bar{\tau}_l) \quad \textit{for } \tau = [\tau_1,\dots,\tau_m,\bar{\tau}_1,\dots,\bar{\tau}_l]_\bullet.$$

Proposition 5.4. *The AVF discrete gradient scheme*

$$\frac{\hat{x}-x}{h}=S\Big(\frac{x+\hat{x}}{2}\Big)\int_0^1 \nabla H((1-\xi)x+\xi\hat{x})\,\mathrm{d}\xi \tag{5.57}$$

is a second order P-series method.

Proof. As in the proof of Proposition 5.2, we define $\hat{e}$ by $\hat{e}(\emptyset)=1$ and $\hat{e}(\tau)=0$ for all $\tau\neq\emptyset$. Now, assume that the solution $\hat{x}$ of (5.57) can be written as the P-series $\hat{x}=P(\Phi,x)$. Then, using Lemma 5.7, we find the P-series

$$\begin{aligned}
&hS\Big(\frac{x+\hat{x}}{2}\Big)\int_0^1 \nabla H((1-\xi)x+\xi\hat{x})\,\mathrm{d}\xi\\
&\qquad\qquad = hS\big(P\big(\frac{1}{2}\hat{e}+\frac{1}{2}\Phi,x\big)\big)\int_0^1 \nabla H\big(P((1-\xi)\hat{e}+\xi\Phi,\,x)\big)\,\mathrm{d}\xi\\
&\qquad\qquad = \int_0^1 hS\big(P\big(\frac{1}{2}\hat{e}+\frac{1}{2}\Phi,\,x\big)\big)\nabla H\big(P((1-\xi)\hat{e}+\xi\Phi,\,x)\big)\,\mathrm{d}\xi\\
&\qquad\qquad = P\Big(\int_0^1 \big(\big(\frac{1}{2}\hat{e}+\frac{1}{2}\Phi\big)\vee\big((1-\xi)\hat{e}+\xi\Phi\big)\big)\,\mathrm{d}\xi,\,x\Big).
\end{aligned}$$

Thus we get $\Phi=\hat{e}+\int_0^1\big(\big(\frac{1}{2}\hat{e}+\frac{1}{2}\Phi\big)\vee\big((1-\xi)\hat{e}+\xi\Phi\big)\big)\,\mathrm{d}\xi=\hat{e}+\int_0^1\big(\big(\frac{1}{2}\Phi\big)\vee\big(\xi\Phi\big)\big)\,\mathrm{d}\xi$. That is, $\Phi(\emptyset)=1$, $\Phi(\bullet)=1$, and

$$\Phi([\tau_1,\ldots,\tau_m,\bar{\tau}_1,\ldots,\bar{\tau}_l])=\frac{1}{(m+1)2^l}\Phi(\tau_1)\cdots\Phi(\tau_m)\Phi(\bar{\tau}_1)\cdots\Phi(\bar{\tau}_l).$$

Writing out the first few terms of the series, we have

$$\begin{aligned}
\hat{x}=x+hf+\frac{h^2}{2}(f^{\circ}f+f^{\bullet}f)+h^3\big(\frac{1}{8}f^{\circ\circ}(f,f)+\frac{1}{4}f^{\circ\bullet}(f,f)+\frac{1}{6}f^{\bullet\bullet}(f,f)\\
+\frac{1}{4}f^{\circ}f^{\circ}f+\frac{1}{4}f^{\circ}f^{\bullet}f+\frac{1}{4}f^{\bullet}f^{\circ}f+\frac{1}{4}f^{\bullet}f^{\bullet}f\big)+\mathcal{O}(h^4),
\end{aligned}$$

which, after comparing with the expanded exact solution (5.52), we see is of order two. □

The following lemma is obtained in a manner similar to Lemma 5.5, i.e. Theorem 2.2 in [23], and hence we present it without its proof.

Lemma 5.8. *Let $P(a,x)$, $P(b,x)$ and $P(c,x)$ be three P-series with $a(\emptyset)=b(\emptyset)=1$ and $c(\emptyset)=0$. Then*

$$hS(P(a,x))\nabla^2 H(P(b,x))P(c,x)=P((a,b)\times c,x)$$

with $((a,b)\times c)(\emptyset) = ((a,b)\times c)(\bullet) = 0$ *and otherwise*

$$((a,b)\times c)(\tau) = \sum_{i=1}^{m} \prod_{j=1, j\neq i}^{m} \prod_{k=1}^{l} a(\bar{\tau}_k) b(\tau_j) c(\tau_i) \tag{5.58}$$

for $\tau = [\tau_1, \dots, \tau_m, \bar{\tau}_1, \dots, \bar{\tau}_l]$.

Note that $\{\emptyset\}$ counts as both a black-rooted and a white-rooted tree. Hence we have e.g.

$((a,b)\times c)($ $) = a(\circ)b(\emptyset)c(\bullet) = a(\bullet)c(\bullet),$

where we also use that $a(\circ) = a(\bullet)$.

We now present a subclass of the AVF discrete gradient method, for which we will find order conditions using Lemma 5.7 and Lemma 5.8. This subclass is every AVF discrete gradient method for which the approximation of $S(x)$ can be written on the form

$$\begin{aligned}\overline{S}(x,\hat{x},h) = \sum_{n=0}^{p-1} h^n \sum_j b_{nj} \Bigg(&\prod_{k=1}^{n} S(\bar{z}_{njk}) \nabla^2 H(z_{njk}) \cdot S(\bar{z}_{nj(n+1)}) \\ &+ (-1)^n S(\bar{z}_{nj(n+1)}) \prod_{k=1}^{n} \nabla^2 H(z_{nj(n-k+1)}) S(\bar{z}_{nj(n-k+1)}) \Bigg),\end{aligned} \tag{5.59}$$

where, if $\hat{x}$ is the solution of

$$\frac{\hat{x}-x}{h} = \overline{S}(x,\hat{x},h)\overline{\nabla}_{\mathrm{AVF}} H(x,\hat{x}),$$

each $z_{njk} := z_{njk}(x,\hat{x},h) = P(\phi_{njk}, x)$ and each $\bar{z}_{njk} := \bar{z}_{njk}(x,\hat{x},h) = P(\psi_{njk}, x)$ can be written as a P-series with $\phi_{njk}(\emptyset) = \psi_{njk}(\emptyset) = 1$ for all n, j, k. We require that $\sum_j b_{0j} = \frac{1}{2}$, which ensures that (5.59) is a consistent approximation of $S(x)$.

Theorem 5.3. *The discrete gradient scheme* (5.9) *with the AVF discrete gradient* (5.10) *and the approximation of* $S(x)$ *given by* (5.59) *is a P-series method.*

Proof. Generalizing the argument in the proof of Proposition 5.4, we find the P-series

$$h S\big(P(a,x)\big) \int_0^1 \nabla H((1-\xi)x + \xi\hat{x})\,\mathrm{d}\xi = P\bigg(\int_0^1 \big(a \vee \big((1-\xi)\hat{e} + \xi\Phi\big)\big)\,\mathrm{d}\xi,\, x \bigg),$$

where $\bar{\theta}(a) := \int_0^1 \big(a \vee \big((1-\xi)\hat{e} + \xi\Phi\big)\big)\,\mathrm{d}\xi = \int_0^1 \big(a \vee \xi\Phi\big)\big)\,\mathrm{d}\xi$, so that $\bar{\theta}(a)(\emptyset) = 0$, $\bar{\theta}(a)(\bullet) = 1$, and

$$\bar{\theta}(a)([\tau_1, \dots, \tau_m, \bar{\tau}_1, \dots, \bar{\tau}_l]) = \frac{1}{m+1} \Phi(\tau_1) \cdots \Phi(\tau_m) a(\bar{\tau}_1) \cdots a(\bar{\tau}_l). \tag{5.60}$$

Thus we may write the solution $\hat{x}$ found from applying the scheme (5.9) with the AVF discrete gradient (5.10) and $\overline{S}(x,\hat{x},h)$ given by (5.59) as

$$\begin{aligned}\hat{x} &= x + \sum_{n=0}^{p-1} h^n \sum_j b_{nj} \Bigg(\prod_{k=1}^{n} S(P(\psi_{njk},x))\nabla^2 H(P(\phi_{njk},x)) \cdot P(\bar{\theta}(\psi_{nj(n+1)}),x) \\ &\quad + (-1)^n \prod_{k=1}^{n} S(P(\psi_{nj(n-k+2)},x))\nabla^2 H(P(\phi_{nj(n-k+1)},x)) \cdot P(\bar{\theta}(\psi_{nj1}),x) \Bigg) \\ &= x + \sum_{n=0}^{p-1} \sum_j b_{nj} \Big(P((\psi_{nj1},\phi_{nj1}) \times \cdots \times (\psi_{njn},\phi_{njn}) \times \bar{\theta}(\psi_{nj(n+1)}),x) \\ &\quad + (-1)^n P((\psi_{nj(n+1)},\phi_{njn}) \times \cdots \times (\psi_{nj2},\phi_{nj1}) \times \bar{\theta}(\psi_{nj1}),x) \Big) \\ &= P(\Phi,x),\end{aligned} \tag{5.61}$$

with

$$\begin{aligned}\Phi =& \hat{e} + \sum_{n=0}^{p-1} \sum_j b_{nj} \big((\psi_{nj1},\phi_{nj1}) \times \cdots \times (\psi_{njn},\phi_{njn}) \times \bar{\theta}(\psi_{nj(n+1)}) \\ &+ (-1)^n (\psi_{nj(n+1)},\phi_{njn}) \times \cdots \times (\psi_{nj2},\phi_{nj1}) \times \bar{\theta}(\psi_{nj1}) \big).\end{aligned} \tag{5.62}$$

□

Theorem 5.4. *The AVF discrete gradient method with* $\overline{S}$ *given by* (5.59) *is of order p if and only if*

$$\Phi(\tau) = \frac{1}{\gamma(\tau)} \quad \textit{for } |\tau| \le p. \tag{5.63}$$

The values $\Phi(\tau)$ can be found from (5.62) using (5.58) recursively and then (5.60). However, a more convenient approach is derived in the next section.

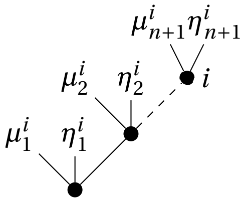
5.4.2 Order conditions

This section is devoted to generalization of the results in Section 5.3.1 to the cases where $S(x)$ is not necessarily constant. To that end, for a tree $\tau \in TP_\bullet$, we cut off all branches between black and white nodes and denote the mono-colored tree we are left with by τ^b. We number the nodes in that tree as before, from 1 to $|\tau^b|$, and reattach the cut-off parts to the tree to get τ again. Let μ denote a forest of black-rooted trees and η a forest of white-rooted trees. Then, for a given node $i \in [1,\ldots,|\tau^b|]$ on level $n+1$, there exists a unique set of forests $\hat{\tau}^i = \{(\mu_1^i,\eta_1^i),\ldots,(\mu_{n+1}^i,\eta_{n+1}^i)\}$ such that

$$\tau = [(\mu_1^i,\eta_1^i)] \circ [(\mu_2^i,\eta_2^i)] \circ \cdots \circ [(\mu_{n+1}^i,\eta_{n+1}^i)].$$

That is,

$\tau =$



Now we can generalize Proposition 5.3 as follows.

Proposition 5.5. *The* Φ *of* (5.62) *can be found by*

$$\Phi(\tau) = \hat{e}(\tau) + \sum_{i=1}^{|\tau^b|} \Lambda(\hat{\tau}^i) \tag{5.64}$$

where $\hat{e}(\emptyset) = 1$ *and* $\hat{e}(\tau) = 0$ *for all* $\tau \neq \emptyset$*, and*

$$\begin{aligned}\Lambda(\hat{\tau}^i) = &\theta([\mu^i_{n+1}]) \sum_j b_{nj} \Big(\psi_{nj1}(\eta^i_1)\phi_{nj1}(\mu^i_1) \cdots \\ &\cdot \psi_{njn}(\eta^i_n)\phi_{njn}(\mu^i_n)\psi_{nj(n+1)}(\eta^i_{n+1}) \\ &+ (-1)^n \psi_{nj(n+1)}(\eta^i_1)\phi_{njn}(\mu^i_1)\psi_{njn}(\eta^i_2) \cdots \\ &\cdot \phi_{nj1}(\mu^i_n)\psi_{nj1}(\eta^i_{n+1}) \Big),\end{aligned} \tag{5.65}$$

with

$$\theta([\tau_1, \ldots, \tau_m]) = \frac{1}{m+1} \Phi(\tau_1) \cdots \Phi(\tau_m).$$

Proof. Defining n_i and C_i as in the proof of Proposition 5.3, we have

$$\begin{aligned}&[(\mu^i_{n_i+1}, \eta^i_{n_i+1})] = [(\mu^k_{n_k}, \eta^k_{n_k})] \circ [(\mu^k_{n_k+1}, \eta^k_{n_k+1})] \qquad \text{for all } k \in C_i, \\ &((a,b) \times c)([(\mu^i_{n_i+1}, \eta^i_{n_i+1})]) = \sum_{k \in C_i} a(\eta^k_{n_k}) b(\mu^k_{n_k}) c([\mu^k_{n_k+1}, \eta^k_{n_k+1}]).\end{aligned}$$

Observe that $\bar{\theta}(a)([\mu, \eta]) = a(\eta)\theta([\mu])$. For $n = 0$ we have

$$\bar{\theta}(\psi_{0j1})(\tau) = \bar{\theta}(\psi_{0j1})([\mu^1_1, \eta^1_1]) = \psi_{0j1}(\eta^1_1)\theta([\mu^1_1]),$$

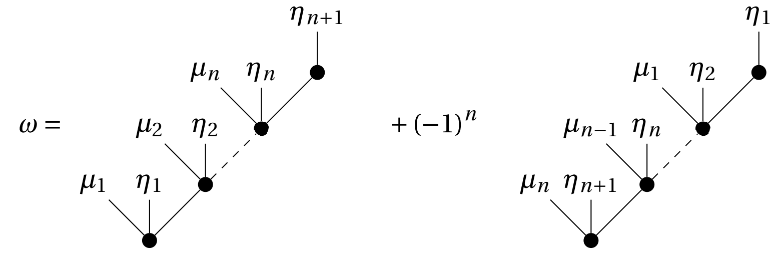
and for $n > 0$ we get

$$
\begin{aligned}
&((\psi_{nj1},\phi_{nj1}) \times \cdots \times (\psi_{njn},\phi_{njn}) \times \bar{\theta}(\psi_{nj(n+1)}))(\tau) \\
&\quad = ((\psi_{nj1},\phi_{nj1}) \times \cdots \times (\psi_{njn},\phi_{njn}) \times \bar{\theta}(\psi_{nj(n+1)}))([\mu_1^1,\eta_1^1]) \\
&\quad = \sum_{i_1 \in C_1} \psi_{nj1}(\eta_1^{i_1})\phi_{nj1}(\mu_1^{i_1})((\psi_{nj2},\phi_{nj2}) \times \cdots \\
&\qquad\qquad \times (\psi_{njn},\phi_{njn}) \times \bar{\theta}(\psi_{nj(n+1)}))([\mu_2^{i_1},\eta_2^{i_1}]) \\
&\qquad\qquad \vdots \\
&\quad = \sum_{i_1 \in C_1} \cdots \sum_{i_n \in C_{i_{n-1}}} \psi_{nj1}(\eta_1^{i_n})\phi_{nj1}(\mu_1^{i_n}) \cdots \\
&\qquad\qquad \cdot \psi_{njn}(\eta_n^{i_n})\phi_{njn}(\mu_n^{i_n})\bar{\theta}(\psi_{nj(n+1)})([\mu_{n+1}^{i_n},\eta_{n+1}^{i_n}]) \\
&\quad = \sum_{i \text{ on level } n+1} \psi_{nj1}(\eta_1^{i})\phi_{nj1}(\mu_1^{i}) \cdots \\
&\qquad\qquad \cdot \psi_{njn}(\eta_n^{i})\phi_{njn}(\mu_n^{i})\psi_{nj(n+1)}(\eta_{n+1}^{i})\theta([\mu_{n+1}^{i}]).
\end{aligned}
$$

Inserting this and the corresponding result for $((\psi_{nj(n+1)},\phi_{njn}) \times \cdots \times (\psi_{nj2},\phi_{nj1}) \times \bar{\theta}(\psi_{nj1}))(\tau)$ in (5.62), we get (5.65). □

Note that if τ only has black nodes, we have $\Lambda(\hat{\tau}^1) = \theta(\tau)\sum_j b_{0j}(\psi_{0j1}(\emptyset) + \psi_{0j1}(\emptyset)) = \theta(\tau)$, and also $\Lambda(\hat{\tau}^i) = 0$ for all nodes i on level 2. Thus (5.64) simplifies to (5.39).

Like for the constant S case, the order conditions can be given for energy-preserving linear combinations of elementary differentials instead for each elementary differential. In the following generalization of Lemma 5.6, we state that the energy-preserving linear combinations of bi-colored rooted trees are given by



Theorem 5.5. *Let $\mu_1, \mu_2, \ldots, \mu_n$ be arbitrary forests of black-rooted trees and $\eta_1, \eta_2, \ldots, \eta_{n+1}$ arbitrary forests of white-rooted trees. Given $f(x) = S(x)\nabla H(x)$, where $S(x)$ is a skew-symmetric matrix, and elementary differentials defined by* (5.56)*, the linear combinations of trees given by*

$$
\begin{aligned}
\omega = {} & [(\mu_1,\eta_1)] \circ \cdots \circ [(\mu_n,\eta_n)] \circ [\eta_{n+1}] \\
& + (-1)^n\, [(\mu_n,\eta_{n+1})] \circ \cdots \circ [(\mu_1,\eta_2)] \circ [\eta_1]
\end{aligned}
\tag{5.66}
$$

are energy-preserving in the sense that $F(\omega)(x)\cdot\nabla H(x) = 0$.

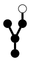
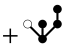
Proof. For any forest of black-rooted trees μ_j, we have $F([\mu_j]\circ[\emptyset]) = SB_jS\nabla H$ for some symmetric matrix B_j, suppressing the argument x. Similarly, for a forest of white-rooted trees η_j, we have $F([\eta_j]) = W_j\nabla H$ for some skew-symmetric matrix W_j. Note that the empty forest is considered both a black-rooted and a white-rooted forest, and accordingly we have $F([\emptyset]\circ[\emptyset]) = F($ $) = S(\nabla^2 H)S\nabla H$ and $F([\emptyset]) = F(\bullet) = S\nabla H$. For these matrices B_j and W_j corresponding to the forests μ_j and η_j, we get

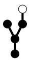
$$F\big([(\mu_1,\eta_1)]\circ\cdots\circ[(\mu_n,\eta_n)]\circ[\eta_{n+1}]\big) = W_1B_1W_2B_2\cdots B_nW_{n+1}\nabla H.$$

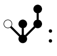
We have

$$(W_1B_1W_2B_2\cdots B_nW_{n+1})^T = \begin{cases} -W_{n+1}B_nW_nB_{n-1}\cdots B_1W_1 & \text{if } n \text{ even,}\\ W_{n+1}B_nW_nB_{n-1}\cdots B_1W_1 & \text{if } n \text{ odd.}\end{cases}$$

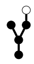
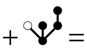
Thus $F(\omega)(x)$ is a skew-symmetric matrix times $\nabla H(x)$, and the statement in the above theorem follows directly. □

Example 5.4. *We show that the combination*  $+$  *is energy-preserving.*


$$: \quad (f^\bullet f^{\bullet\bullet}(f,f^\circ f))^i = S^i_j g^j_k S^k_l g^l_{mo} S^m_n g^n S^o_{pq} g^p S^q_r g^r$$
$$= S^i_j g^j_k S^k_l g^l_{mo} S^m_n g^n S^o_{pq} S^q_r g^r g^p.$$


$$: \quad (f^{\circ\bullet\bullet}(f,f,f^\bullet f))^i = S^i_{jk} g^j_{mo} S^k_l g^l S^m_n g^n S^o_p g^p_q S^q_r g^r$$
$$= S^i_{jk} S^k_l g^l g^j_{mo} S^m_n g^n S^o_r g^r_q S^q_p g^p.$$

For this linear combination on the form (5.66)*, we have* $\eta_1 = \eta_2 = \emptyset, \eta_3 = \circ$, $\mu_1 = \emptyset, \mu_2 = \bullet$, *with the corresponding matrices* $W_1 = W_2 = S, (W_3)^i_j = S^i_{jk}S^k_l g^l$ *and* $B_1 = \nabla^2 H, (B_2)^j_m = g^j_{km}S^k_l g^l$. *Thus we get*

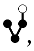
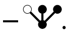
 $+$ 
$$= f^\bullet f^{\bullet\bullet}(f,f^\circ f) + f^{\circ\bullet\bullet}(f,f,f^\bullet f) = Z\nabla H,$$

where $Z := S(\nabla^2 H)SB_2W_3 + W_3B_2S(\nabla^2 H)S$ *is a skew-symmetric matrix.*

For bi-colored trees, we define a node on the tree τ to be a leaf if it is a leaf on the corresponding cut tree τ^b by the definition of leaves given in the previous chapter. We let I_l be the set of leaves and I_n the set of non-leaf nodes which are also in τ^b, so that $I_l \cup I_n = [1,\dots,|\tau^b|]$. In contrast to the case with mono-colored trees, a leaf i on level one or two of a bi-colored tree may give rise to a non-zero energy-preserving linear combination; it does so if and only if $\eta^i_k \neq \emptyset$ for any $k = 1,2$. Accordingly, $\Lambda(\hat{\tau}^i)$ is calculated in (5.64) also when $n = 0,1$. Furthermore, two leaves i and j on the same level will belong to two different energy-preserving combinations if $\eta^i_{n+1} \neq \eta^j_{n+1}$. Therefore we now simply state that a tree with r leaves, also including the lower two levels, belong to at most r non-zero linear combinations. We thus get r terms on the left hand side of

$$\sum_{i \in I_l} \Lambda(\hat{\tau}^i) = \frac{1}{\gamma(\tau)} - \hat{e}(\tau) - \sum_{i \in I_n} \frac{\Lambda(\{(\mu^i_1, \eta^i_1), \dots, (\mu^i_n, \eta^i_n), (\emptyset, \eta^i_{n+1})\})}{(|\mu^i_{n+1}| + 1)\gamma(\mu^i_{n+1})}, \tag{5.67}$$

which is equivalent to (5.63) if we assume the conditions for lower order to be satisfied.

Example 5.5. *Consider the tree* $\tau =$ , *which is part of the energy-preserving linear combination* $-$ . *Assume that the order conditions up to and including order three are all satisfied. The cut tree* $\tau^b =$ *has three nodes of which two are leaves. Node number* 2 *is a leaf on level* 2 *with* $\eta^2_1 = \eta^2_2 = \emptyset$, *and thus gives* $\Lambda(\hat{\tau}^2) = 0$. *We find for the other two,*

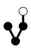
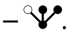
$$\begin{aligned}
\Lambda(\hat{\tau}^1) &= \Lambda(\{((\bullet, \overset{\circ}{\bullet}), \emptyset)\}) = \frac{1}{(|\mu^1_1| + 1)\gamma(\mu^1_1)} \Lambda\big(\{(\emptyset, \emptyset)\}\big) = \frac{1}{(2+1)\gamma(\bullet)\gamma(\overset{\circ}{\bullet})} \frac{1}{\gamma(\bullet)} = \frac{1}{6}, \\
\Lambda(\hat{\tau}^3) &= \Lambda(\{(\bullet, \emptyset), (\emptyset, \circ)\}) = \sum_j b_{1j} (\phi_{1j1}(\bullet)\psi_{1j2}(\circ) - \psi_{1j1}(\circ)\phi_{1j1}(\bullet)) \\
&= \sum_j b_{1j} \phi_{1j1}(\bullet)(\psi_{1j2} - \psi_{1j1})(\bullet).
\end{aligned}$$

For the right hand side of (5.67) *we get*

$$\frac{1}{\gamma(\tau)} - \Lambda(\hat{\tau}^1) = \frac{1}{8} - \frac{1}{6} = -\frac{1}{24},$$

and thus the order condition

$$\sum_j b_{1j} \phi_{1j1}(\bullet)(\psi_{1j2} - \psi_{1j1})(\bullet) = -\frac{1}{24}$$

for the energy-preserving linear combination  $-$ .

Even though the number of black-rooted bi-colored trees grows very quickly, e.g. to 26 for $|\tau| = 4$ and 107 for $|\tau| = 5$, finding and satisfying the order conditions is not as daunting a task as it might first appear. First of all, it suffices to find order conditions for the non-zero linear combinations given by (5.66). Moreover, a couple key observations simplifies the process further:

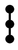
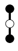
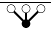
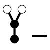
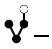
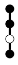
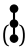
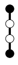
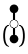
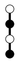
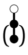
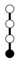
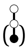
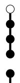
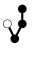
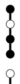
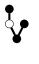
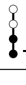
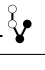
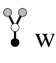
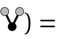
$\lvert\tau\rvert$	ω	Order condition
1	$\bullet$	$2\sum_j b_{0j} = 1$
2		$2\sum_j b_{0j}\psi_{0j1}(\bullet) = \frac{1}{2}$
3		$2\sum_j b_{0j}\psi_{0j1}(\bullet)^2 = \frac{1}{3}$
		$\sum_j b_{2j} = -\frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($ $) = \frac{1}{6}$
		$2\sum_j b_{0j}\psi_{0j1}($ $) = \frac{1}{6}$
	$-$	$\sum_j b_{1j}(\psi_{1j2}-\psi_{1j1})(\bullet) = -\frac{1}{12}$
4		$2\sum_j b_{0j}\psi_{0j1}(\bullet)^3 = \frac{1}{4}$
		$2\sum_j b_{2j}\psi_{2j2}(\bullet) = -\frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($ $) = \frac{1}{12}$
		$2\sum_j b_{0j}\psi_{0j1}($ $) = \frac{1}{12}$
		$2\sum_j b_{0j}\psi_{0j1}($ $) = \frac{1}{12}$
		$2\sum_j b_{0j}\psi_{0j1}(\bullet)\psi_{0j1}($ $) = \frac{1}{8}$
		$2\sum_j b_{0j}\psi_{0j1}(\bullet)\psi_{0j1}($ $) = \frac{1}{8}$
	$+$	$\sum_j b_{2j}(\phi_{2j1}+\phi_{2j2})(\bullet) = -\frac{1}{24}$
	 $-$	$\sum_j b_{1j}(\psi_{1j2}(\bullet)^2-\psi_{1j1}(\bullet)^2) = -\frac{1}{12}$
	 $-$	$\sum_j b_{1j}\phi_{1j1}(\bullet)(\psi_{1j1}-\psi_{1j0})(\bullet) = -\frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($$) = \frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($$) = \frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($$) = \frac{1}{24}$
		$2\sum_j b_{0j}\psi_{0j1}($$) = \frac{1}{24}$
	 $+$ 	$\sum_j b_{2j}(\psi_{2j1}+\psi_{2j3})(\bullet) = -\frac{1}{24}$
	 $-$ 	$\sum_j b_{1j}(\psi_{1j2}-\psi_{1j1})($ $) = -\frac{1}{24}$
	 $-$ 	$\sum_j b_{1j}(\psi_{1j2}-\psi_{1j1})($ $) = -\frac{1}{24}$

Table 5.4: Linear combinations ω of bi-colored black-rooted trees corresponding to energy-preserving elementary differentials of $f(x) = S(x)\nabla H(x)$, where $S(x)$ is a skew-symmetric matrix, as well as their associated order conditions for the discrete gradient method (5.9) with the AVF discrete gradient (5.10) and $\overline{S}(x,\hat{x},h)$ given by (5.59).

- The large number of trees τ for which $\tau^b = \bullet$, i.e. trees with no black nodes on level 2, are all energy-preserving. They can be written $\tau = [\eta_1^1]$, and their order condition is given by

$$2\sum_j b_{0j}\psi_{0j1}(\eta_1^1) = \frac{1}{\gamma(\tau)}.$$

- For trees that are identical except for the colors of the descendants of white nodes, it suffices to calculate one order condition. E.g. for  we have the order condition $2b_{0j}\psi_{0j1}($$) = \frac{1}{12}$, where each of the gray nodes may be black or white. To satisfy these conditions, it is natural to require that $\bar{z}_{0j1}$ in (5.59) is a B-series *up to* order $p-1$.

From the order conditions displayed in Table 5.4 we find that one second order scheme is given by (5.9) using the AVF discrete gradient (5.10) and an explicit skew-symmetric approximation of S given by $\overline{S}(x,\cdot,h) = S(x+\frac{1}{2}hf(x))$. A third order scheme is obtained if we instead use the skew-symmetric approximation of S explicitly given by

$$\begin{aligned}\overline{S}(x,\cdot,h) = &\frac{1}{4}S(x) + \frac{3}{4}S(z_2)\\ &+\frac{1}{4}h\big(S(z_1)\nabla^2 H(x)S(x) - S(x)\nabla^2 H(x)S(z_1)\big)\\ &-\frac{1}{12}h^2\, S(x)\nabla^2 H(x)S(x)\nabla^2 H(x)S(x),\end{aligned} \tag{5.68}$$

where $z_1 = x+\frac{1}{3}hf(x)$, $z_2 = x+\frac{2}{3}hf(z_1)$.

A symmetric fourth order scheme is given by (5.9) using the AVF discrete gradient (5.10) and the skew-symmetric approximation of S

$$\begin{aligned}\overline{S}(x,\hat{x},h) = &\frac{1}{2}S\Big(\bar{x} - \frac{1}{\sqrt{12}}hf\big(\bar{x}+\frac{1}{\sqrt{12}}hf(\bar{x})\big)\Big)\\ &+\frac{1}{2}S\Big(\bar{x} + \frac{1}{\sqrt{12}}hf\big(\bar{x}-\frac{1}{\sqrt{12}}hf(\bar{x})\big)\Big)\\ &+\frac{1}{2}hS\big(\bar{x}+\frac{1}{12}hf(\bar{x})\big)\nabla^2 H(\bar{x})S\big(\bar{x}-\frac{1}{12}hf(\bar{x})\big)\\ &-\frac{1}{2}hS\big(\bar{x}-\frac{1}{12}hf(\bar{x})\big)\nabla^2 H(\bar{x})S\big(\bar{x}+\frac{1}{12}hf(\bar{x})\big)\\ &-\frac{1}{12}h^2\, S(\bar{x})\nabla^2 H(\bar{x})S(\bar{x})\nabla^2 H(\bar{x})S(\bar{x}),\end{aligned} \tag{5.69}$$

where $\bar{x} = \frac{x+\hat{x}}{2}$. Another fourth order scheme is obtained if we instead use the explicit skew-symmetric approximation of S found by

$$\begin{aligned}\overline{S}(x,\cdot,h) &= \frac{1}{2}(S(z_5+z_6)+S(z_5-z_6))\\ &\quad+\frac{1}{12}h\big(S(z_2)\nabla^2 H(z_1)S(x)-S(x)\nabla^2 H(z_1)S(z_2)\big)\\ &\quad-\frac{1}{12}h^2\, S(z_1)\nabla^2 H(z_1)S(z_1)\nabla^2 H(z_1)S(z_1),\end{aligned} \tag{5.70}$$

where

$$\begin{gathered} z_1 = x+\frac{1}{2}hf(x), \quad z_2 = x+hf(z_1), \quad z_3 = x+hf(z_2), \quad z_4 = x+hf(z_3),\\ z_5 = \frac{1}{3}(x+z_1+z_2)+\frac{1}{12}(-z_3+z_4), \qquad z_6 = \frac{\sqrt{3}}{36}(7x-2z_1-4z_2+z_3-2z_4).\end{gathered}$$

5.5 Order conditions for general discrete gradient methods

We will now generalize the results of the two previous chapters to discrete gradient methods with a general discrete gradient, as defined by (5.7)–(5.8). To that end, we introduce two new series in the vein of B- and P-series, as well as related tree structures.

5.5.1 The constant S case

Consider mono-colored rooted trees whose nodes can have two different shapes: the circle shape of the nodes in trees of B-series, but also a triangle shape. Let TG be the set of such trees whose leaves are always circles. That is, from the first tree $\bullet$, every tree $\tau \in TG$ can be built recursively through

$$[\tau_1,\ldots,\tau_m]_\bullet, \quad [\tau_1,\ldots,\tau_m]_\blacktriangle, \quad \tau_1,\ldots,\tau_m \in TG,$$

which denotes the grafting of the trees $\tau_1,\ldots,\tau_m$ to a root $\bullet$ or $\blacktriangle$, respectively. The elementary differentials $F(\tau)$ corresponding to a tree $\tau \in TG$ are likewise defined recursively by $F(\bullet)(x) = f(x) = S\nabla H(x)$ and

$$F(\tau)(x) = \begin{cases} SD^m\nabla H(x)(F(\tau_1)(x),\ldots,F(\tau_m)(x)) & \text{for } \tau = [\tau_1,\ldots,\tau_m]_\bullet,\\ SD_2^{m-1}Q(x,x)(F(\tau_1)(x),\ldots,F(\tau_m)(x)) & \text{for } \tau = [\tau_1,\ldots,\tau_m]_\blacktriangle.\end{cases}$$

We can then define a generalization of B-series which includes these elementary differentials.

Definition 5.4. A G-series is a formal series of the form

$$G(\phi, x) = \phi(\emptyset)x + \sum_{\tau \in TG} \frac{h^{|\tau|}}{\sigma(\tau)} \phi(\tau) F(\tau)(x), \tag{5.71}$$

where $\phi : TG \cup \{\emptyset\} \to \mathbb{R}$ is an arbitrary mapping, and the symmetry coefficient σ is given by (5.31).

The G-series of the exact solution is given by $x(t_0 + h) = G(\xi, x(t_0))$, with

$$\xi(\tau) = \begin{cases} \frac{1}{\gamma(\tau)} & \text{if } \tau \in T, \\ 0 & \text{otherwise.} \end{cases} \tag{5.72}$$

For use in the remainder of this paper, we generalize the Butcher product by the definition

$$u \circ v = [u_1, \dots, u_m, v]_\star, \quad \text{for } u = [u_1, \dots, u_m]_\star, \quad \star \in \{\bullet, \blacktriangle\}.$$

Furthermore, we let $|\tau|$ denote the total number of nodes in τ, and $|\tau|_\star$ the number of nodes of type $\star$. Let SG be the set of tall trees in TG; that is, the set of threes with only one node on each level. For a tree $\tau \in TG$, number every tree from 1 to $|\tau|$, as before. For any node i on level $n+1$, we define the stem $s^i \in SG$ to be the tall tree consisting of the nodes connecting the root to node i, including the root and node i. Denote the j^{th} node of s^i by s^i_j, so that s^i_1 is the root and $s^i_{n+1} = i$. Then we have a unique set of forests $\hat{\tau}^i = \{\mu^i_1, \dots, \mu^i_{n+1}\}$ such that

$$\tau = [\mu^i_1]_{s^i_1} \circ [\mu^i_2]_{s^i_2} \circ \dots \circ [\mu^i_{n+1}]_{s^i_{n+1}}.$$

That is,

$$\tau = \begin{array}{ccc} & & \mu^i_{n+1} \\ & & | \\ & \mu^i_2 & s^i_{n+1} \\ & \backslash & / \\ \mu^i_1 & s^i_2 & \\ \backslash & / & \\ s^i_1 & & \end{array}$$

The following lemma is a generalization of Lemma 5.5 to G-series. Its proof is very similar to the proof of [23, Theorem 2.2], and hence omitted.

Lemma 5.9. *Let $G(a, x)$ and $G(b, x)$ be two G-series with $a(\emptyset) = 1$ and $b(\emptyset) = 0$. Then the G-series $hS\nabla^2 H(G(a, x))G(b, x) = G(a \times b, x)$ is given by $(a \times b)(\emptyset) = (a \times b)(\bullet) = 0$ and otherwise*

$$(a \times b)(\tau) = \begin{cases} \sum_{i=1}^m \prod_{j=1, j \neq i}^m a(\tau_j) b(\tau_i) & \textit{for } \tau = [\tau_1, \dots, \tau_m]_\bullet, \\ 0 & \textit{for } \tau = [\tau_1, \dots, \tau_m]_\blacktriangle. \end{cases}$$

Moreover, $hSQ(x, G(a,x))G(b,x) = G(a \otimes b, x)$*, with* $(a \otimes b)(\emptyset) = (a \otimes b)(\bullet) = 0$ *and otherwise*

$$(a \otimes b)(\tau) = \begin{cases} 0 & \textit{for } \tau = [\tau_1, \dots, \tau_m]_\bullet, \\ \sum_{i=1}^m \prod_{j=1, j\neq i}^m a(\tau_j) b(\tau_i) & \textit{for } \tau = [\tau_1, \dots, \tau_m]_\blacktriangle. \end{cases}$$

To every stem $s \in SG$ of height $n+1 = |s|$, we associate coefficients b_{sj} and ϕ_{sjk}. Letting s_k be the k^{th} node of s, we define the function

$$R(\phi_{sjk}, x) := \begin{cases} \nabla^2 H(G(\phi_{sjk}, x)) & \text{if } s_k = \bullet, \\ Q(x, G(\phi_{sjk}, x)) & \text{if } s_k = \blacktriangle. \end{cases}$$

Then we have $hSR(\phi_{sjk}, x)G(b,x) = G(\phi_{sjk} \diamond b)$, with $(\phi_{sjk} \diamond b)(\emptyset) = (\phi_{sjk} \diamond b)(\bullet) = 0$ and

$$(\phi_{sjk} \diamond b)(\tau) = \begin{cases} \sum_{i=1}^m \prod_{j=1, j\neq i}^m \phi_{sjk}(\tau_j) b(\tau_i) & \text{for } \tau = [\tau_1, \dots, \tau_m]_{s_k}, \\ 0 & \text{if root of } \tau \neq s_k. \end{cases}$$

Consider now the class of skew-symmetric and consistent approximations to S that can be written on the form

$$\overline{S}(x,y,h) = \sum_{s \in SG} h^n \sum_j b_{sj} \left(\prod_{k=1}^n SR(\phi_{sjk}, x) + (-1)^{|s|_\bullet - 1} \prod_{k=1}^n SR(\phi_{sj(n-k+1)}, x) \right) S \tag{5.73}$$

whenever y is the solution of

$$\frac{y - x}{h} = \overline{S}(x,y,h) \overline{\nabla} H(x,y),$$

with $\phi_{sjk}(\emptyset) = 1$ for every s, j, k, and with $\sum_j b_{\bullet j} = \frac{1}{2}$.

Lemma 5.10. *The discrete gradient method* (5.9) *with* $\overline{S}(x, \hat{x}, h)$ *given by* (5.73) *and* $\overline{\nabla} H \in C^\infty(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$ *is a G-series method when applied to a constant S skew-gradient system* (5.6).

Proof. Assume that the solution $\hat{x}$ of (5.9) with $\overline{S}(x, \hat{x}, h)$ given by (5.73) can be written as the G-series $\hat{x} = G(\Phi, x)$. Then, using Lemma 5.2 and $\overline{\nabla} H(x,x) = \nabla H(x)$,

$$\begin{aligned} hS\overline{\nabla} H(x, \hat{x}) &= hS \sum_{m=0}^\infty \frac{1}{m!} D_2^m \overline{\nabla} H(x,x)(G(\Phi,x) - x)^m \\ &= hS \sum_{m=0}^\infty \frac{1}{(m+1)!} D^m \nabla H(x)(G(\Phi,x) - x)^m \\ &\quad - hS \sum_{m=1}^\infty \frac{2m}{(m+1)!} D_2^{m-1} Q(x,x)(G(\Phi,x) - x)^m. \end{aligned}$$

Arguing as in the proof of Lemma III.1.9 in [13], we get $hS\overline{\nabla}H(x,\hat{x}) = G(\theta,x)$, with $\theta(\emptyset) = 0$, $\theta(\bullet) = 1$, and

$$\begin{aligned} \theta([\tau_1,\dots,\tau_m]_\bullet) &= \frac{1}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m), \\ \theta([\tau_1,\dots,\tau_m]_\blacktriangle) &= \frac{-2m}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m). \end{aligned} \tag{5.74}$$

Then we can write (5.9) with $\overline{S}(x,\hat{x},h)$ given by (5.73) as

$$\begin{aligned} \hat{x} &= x + \sum_{s\in SG} h^n \sum_j b_{sj}\left(\prod_{k=1}^n SR(\phi_{sjk},x) + (-1)^{|s|_\bullet-1}\prod_{k=1}^n SR(\phi_{sj(n-k+1)},x)\right) G(\theta,x) \\ &= x + G(\theta,x) + \sum_{s\in SG,\ n>0} \sum b_j^s\big(G(\phi_{sj1}\diamond\cdots\diamond\phi_{sjn}\diamond\theta,x) \\ &\qquad\qquad + (-1)^{|s|_\bullet-1}G(\phi_{sjn}\diamond\cdots\diamond\phi_{sj1}\diamond\theta,x)\big) \\ &= G(\Phi,x), \end{aligned}$$

with

$$\Phi = \hat{e} + \theta + \sum_{s\in SG,\ n>0}\sum_j b_{sj}\left(\phi_{sj1}\diamond\cdots\diamond\phi_{sjn}\diamond\theta + (-1)^{|s|_\bullet-1}\phi_{sjn}\diamond\cdots\diamond\phi_{sj1}\diamond\theta\right). \tag{5.75}$$

□

Theorem 5.6. *The discrete gradient method* (5.9) *with* $\overline{S}(x,\hat{x},h)$ *given by* (5.73) *and* $\overline{\nabla}H \in C^\infty(\mathbb{R}^d\times\mathbb{R}^d,\mathbb{R}^d)$ *is of order* p *if and only if*

$$\Phi(\tau) = \xi(\tau) \qquad for\ |\tau| \le p, \tag{5.76}$$

where Φ *is given by* (5.75) *and the* ξ *is given by* (5.72).

We remark that $\overline{\nabla}H \in C^\infty(\mathbb{R}^d\times\mathbb{R}^d,\mathbb{R}^d)$ is a necessary condition for the method to be a G-series method for all S and H, but not for its order; $\overline{\nabla}H \in C^{p-1}(\mathbb{R}^d\times\mathbb{R}^d,\mathbb{R}^d)$ is sufficient for the scheme to be of order p. The following proposition is presented without its proof, which follows along the lines of the proof of Proposition 5.3.

Proposition 5.6. *The* Φ *of* (5.76) *satisfies*

$$\Phi(\tau) = \hat{e}(\tau) + \theta(\tau) + \sum_{i\ s.t.\ n\ge1}\Lambda(\hat{\tau}^i,s^i) \tag{5.77}$$

where $\hat{e}(\emptyset) = 1$ *and* $\hat{e}(\tau) = 0$ *for all* $\tau \neq \emptyset$, θ *is given by* (5.74), *and*

$$\begin{aligned} \Lambda(\hat{\tau}^i,s^i) = \theta([\mu^i_{n+1}]_{s^i_{n+1}})\big(&\sum_j b_{s^ij}\phi_{s^ij1}(\mu^i_1)\cdots\phi_{s^ijn}(\mu^i_n) \\ &+ (-1)^{|s^i|_\bullet-1}\sum_j b_{\hat{s}^ij}\phi_{\hat{s}^ijn}(\mu^i_1)\cdots\phi_{\hat{s}^ij1}(\mu^i_n)\big), \end{aligned} \tag{5.78}$$

with $\hat{s}^i$ *given by* $\hat{s}^i_k = s^i_{n-k+1}$ *for* $k = 1,\dots,n$, *and* $\hat{s}^i_{n+1} = s^i_{n+1}$.

As for the AVF method, one does not need to find the order conditions for every tree; it suffices to find the order condition for each energy-preserving linear combination of the form

$$\begin{aligned}\omega &= [\mu_1]_{s_1} \circ [\mu_2]_{s_2} \circ \cdots [\mu_n]_{s_n} \circ [\emptyset]_\bullet \\ &\quad + (-1)^n [\mu_n]_{s_n} \circ [\mu_{n-1}]_{s_{n-1}} \circ \cdots [\mu_1]_{s_1} \circ [\emptyset]_\bullet.\end{aligned} \tag{5.79}$$

The above does not give every energy-preserving linear combination of the elementary differentials of G-series; it gives the combinations one gets in the scheme (5.9) with $\overline{S}(x,\hat{x},h)$ given by (5.73). Now, let again I_l and I_n denote the sets of leaf nodes and non-leaf nodes, respectively. If we assume the conditions for order $< p$ to be satisfied, we have an equivalent order condition to (5.76) by

$$\sum_{i\in I_l} \Lambda(\hat{\tau}^i, s^i) = \xi(\tau) - \hat{e}(\tau) - \sum_{i\in I_n} \Lambda(\hat{\tau}^i, s^i), \tag{5.80}$$

where we may use the relation

$$\Lambda(\{\mu_1^i,\ldots,\mu_n^i,\mu_{n+1}^i\}, s^i) = \hat{\theta}([\mu_{n+1}^i]_{s_{n+1}^i})\Lambda(\{\mu_1^i,\ldots,\mu_n^i,\emptyset\},\bar{s}^i)$$

to calculate $\Lambda(\hat{\tau}^i)$ for $i \in I_n$. Here $\bar{s}^i$ is s^i with s_{n+1}^i replaced by $\bullet$, and $\hat{\theta}(\emptyset) = 0$, $\hat{\theta}(\bullet) = 1$, and

$$\begin{aligned}\hat{\theta}([\tau_1,\ldots,\tau_m]_\bullet) &= \frac{1}{m+1}\xi(\tau_1)\cdots\xi(\tau_m), \\ \hat{\theta}([\tau_1,\ldots,\tau_m]_\blacktriangle) &= \frac{-2m}{m+1}\xi(\tau_1)\cdots\xi(\tau_m).\end{aligned} \tag{5.81}$$

Note that $\Lambda(\hat{\tau}^1, s^1) = \hat{\theta}(\tau)$.

Example 5.6. *Consider* $\tau = [\bullet,[\bullet]_\blacktriangle]_\blacktriangle$*, which is part of two combinations of the form* (5.79)*:* $\omega = [\bullet,[\bullet]_\blacktriangle]_\blacktriangle + [[\bullet,\bullet]_\blacktriangle]_\blacktriangle$ *and* $\omega = 2[\bullet,[\bullet]_\blacktriangle]_\blacktriangle$*. We calculate*

$$\begin{aligned}
\Lambda(\hat{\tau}^1, s^1) &= \Lambda(\{(\bullet,[\bullet]_\blacktriangle)\},\blacktriangle) = \hat{\theta}([\bullet,[\bullet]_\blacktriangle]_\blacktriangle) = 0,\\
\Lambda(\hat{\tau}^2, s^2) &= \Lambda(\{[\bullet]_\blacktriangle,\emptyset\},[\bullet]_\blacktriangle) = \sum_j b_{s^2j}\phi_{s^2j1}([\bullet]_\blacktriangle) + \sum_j b_{\hat{s}^2j}\phi_{\hat{s}^2j1}([\bullet]_\blacktriangle) = 2\sum_j b_{s^2j}\phi_{s^2j1}([\bullet]_\blacktriangle)\\
\Lambda(\hat{\tau}^3, s^3) &= \Lambda(\{\bullet,\bullet\},[\blacktriangle]_\blacktriangle) = \hat{\theta}([\bullet]_\blacktriangle)\Lambda(\{\bullet,\emptyset\},[\bullet]_\blacktriangle) = \hat{\theta}([\bullet]_\blacktriangle)(-\frac{1}{2}\hat{\theta}([\bullet,\bullet]_\blacktriangle)) = -1(-\frac{1}{2}(-\frac{4}{3})) = -\frac{2}{3},\\
\Lambda(\hat{\tau}^4, s^4) &= \Lambda(\{\bullet,\emptyset,\emptyset\},[[\bullet]_\blacktriangle]_\blacktriangle) = \sum_j b_{s^4j}\phi_{s^4j1}(\bullet) + \sum_j b_{\hat{s}^4j}\phi_{\hat{s}^4j2}(\bullet) = \sum_{j,k} b_{s^4j}\phi_{s^4jk}(\bullet).
\end{aligned}$$

Thus (5.80) *becomes*

$$2\sum_j b_{s^2j}\phi_{s^2j1}([\bullet]_\blacktriangle) + \sum_{j,k} b_{s^4j}\phi_{s^4jk}(\bullet) = \frac{2}{3}$$

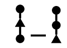
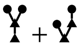
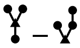
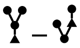
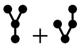
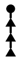
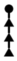
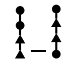
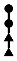
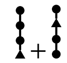
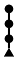
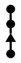
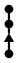
$\lvert\tau\rvert$	ω	s	Order condition
1	$\bullet$	$\bullet$	$\sum_j b_{sj} = \frac{1}{2}$
2			$\sum_j b_{sj} = \frac{1}{2}$
3			$\sum_j b_{sj}\phi_{sj1}(\bullet) = \frac{1}{3}$
			$\sum_j b_{sj} = \frac{1}{2}$
			$\sum_j b_{sj} - \sum_j b_{\bar{s}j} = 0$
			$\sum_j b_{sj} = -\frac{1}{24}$
4			$\sum_j b_{sj}\phi_{sj1}(\bullet)^2 = \frac{1}{4}$
			$\sum_j b_{sj}\phi_{sj1}($ $) = 0$
			$\sum_j b_{sj}\phi_{sj1}($ $) = \frac{1}{6}$
			$\sum_{j,k} b_{sj}\phi_{sjk}(\bullet) = \frac{2}{3}$
			$\sum_j b_{sj}\phi_{sj2}(\bullet) - \sum_j b_{\bar{s}j}\phi_{\bar{s}j1}(\bullet) = 0$
			$\sum_j b_{sj}\phi_{sj2}(\bullet) - \sum_j b_{\bar{s}j}\phi_{\bar{s}j1}(\bullet) = 0$
			$\sum_{j,k} b_{sj}\phi_{sjk}(\bullet) = -\frac{1}{24}$
			$\sum_j b_{sj} = \frac{1}{2}$
			$\sum_j b_{sj} - \sum_j b_{\bar{s}j} = 0$
			$\sum_j b_{sj} - \sum_j b_{\bar{s}j} = 0$
			$\sum_j b_{sj} = 0$

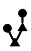
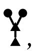
Table 5.5: Energy-preserving linear combinations of the form (5.79) and their associated order conditions for the discrete gradient method (5.9) with $\overline{S}(x,\hat{x},h)$ given by (5.73).

for $\tau =$ *. We do similar calculations for* *, and get* (5.80) *for that to be*

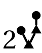
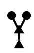
$$2\sum_{j,k} b_{s^4 j}\phi_{s^4 jk}(\bullet) = \frac{4}{3}.$$

Thus we have the order condition

$$\sum_{j,k} b_{s^4 j}\phi_{s^4 jk}(\bullet) = \frac{2}{3} \tag{5.82}$$

for $\omega =$  $+$ *, and*

$$\sum_{j} b_{s^2 j}\phi_{s^2 j1}(\text{ }) = 0$$

for $\omega = 2$*. Note that although the tree*  *gives an energy-preserving elementary differential, this by itself is not of the form* (5.79).

From the order conditions in Table 5.5, we can find an $\overline{S}(x,y,h)$ so that (5.9) becomes a fourth order scheme for any $\overline{\nabla}H \in C^3(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$. For instance, the stem $s =$ has the related order conditions $\sum_j b_{sj} = \frac{1}{2}$ and $\sum_{j,k} b_{sj}\phi_{sjk}(\bullet) = \frac{2}{3}$, which sets the requirements for the term

$$h^2 \sum_j b_{sj}(SQ(x,z_{1j})SQ(x,z_{2j}) + SQ(x,z_{2j})SQ(x,z_{1j}))S.$$

Choosing $b_{s1} = \frac{1}{2}$ and $z_{11} = z_{21} = x + \frac{2}{3}hf(x)$, we have fulfilled these conditions. Likewise, finding terms that satisfy the other order conditions, we get an approximation of S that ensures fourth order convergence, like the $\overline{S}(x,y,h)$ given by (5.29).

5.5.2 The general case

Allowing for S to be a function of the solution, we define now the set TV of bi-colored trees whose nodes are either circles of triangles, and whose leaves on the cut tree τ^b, defined as the mono-colored tree left when all branches between black and white nodes are cut off, are always circles. Denoting as before black-rooted subtrees by τ_i and white-rooted subtrees by $\bar{\tau}_i$, the elementary differentials of trees $\tau \in TV$ are given $F(\bullet)(x) = F(\circ)(x) = f(x) = S\nabla H(x)$ and

$$F(\tau)(x) = \begin{cases} S^{(l)}D^m\nabla H(x)(F(\tau_1)(x),\ldots,F(\bar{\tau}_l)(x)) \\ \qquad\qquad\qquad\qquad \text{for } \tau = [\tau_1,\ldots,\tau_m,\bar{\tau}_1,\ldots,\bar{\tau}_l,]_{\circledcirc}, \\ S^{(l)}D_2^{m-1}Q(x,x)(F(\tau_1)(x),\ldots,F(\bar{\tau}_l)(x)) \\ \qquad\qquad\qquad\qquad \text{for } \tau = [\tau_1,\ldots,\tau_m,\bar{\tau}_1,\ldots,\bar{\tau}_l,]_{\triangle}, \end{cases}$$

where ◉ can be either • or ○ and ◭ can be either ▲ or △. Let $TV_\bullet$ denote the set of trees in TV with black roots, either of the shape • or ▲.

Definition 5.5. A V-series is a formal series of the form

$$V(\phi,x)=\phi(\emptyset)x+\sum_{\tau\in TV_\bullet}\frac{h^{|\tau|}}{\sigma(\tau)}\phi(\tau)F(\tau)(x), \tag{5.83}$$

where $\phi\colon TV_\bullet\cup\{\emptyset\}\to\mathbb{R}$ is an arbitrary mapping, and the symmetry coefficient σ is given by (5.31).

Proofs of the theorems in this section can be obtained similarly to the proofs in Chapter 5.4 and Section 5.5.1, and are therefore omitted.

We consider now approximations of $S(x)$ that can be written as

$$\begin{aligned}\overline{S}(x,y,h)=\sum_{s\in SG}h^n\sum_j b_{sj}\Bigg(&\prod_{k=1}^{n}S(V(\psi_{sjk},x))R(\phi_{sjk},x)\cdot S(V(\psi_{sj(n+1)},x))\\ &+(-1)^{|s|_\bullet-1}S(V(\psi_{sj(n+1)},x))\prod_{k=1}^{n}R(\phi_{sj(n-k+1)},x)S(V(\psi_{sj(n-k+1)},x))\Bigg)\end{aligned} \tag{5.84}$$

whenever y is the solution of

$$\frac{y-x}{h}=\overline{S}(x,y,h)\overline{\nabla}H(x,y),$$

with $\phi_{sjk}(\emptyset)=\psi_{sjk}(\emptyset)=1$ for every s,j,k, and with $\sum_j b_{\bullet j}=\frac{1}{2}$.

Theorem 5.7. *The discrete gradient scheme* (5.9) *with the approximation of* $S(x)$ *given by* (5.84) *and* $\overline{\nabla}H\in C^\infty(\mathbb{R}^d\times\mathbb{R}^d,\mathbb{R}^d)$ *is a V-series method. It can be written* $\hat{x}=V(\Phi,x)$, *with*

$$\begin{aligned}\Phi=\hat{e}+\sum_{s\in SG}\sum_j b_{sj}\big(&(\psi_{sj1},\phi_{sj1})\diamond\cdots\diamond(\psi_{sjn},\phi_{sjn})\diamond\hat{\theta}(\psi_{sj(n+1)})\\ &+(-1)^n(\psi_{sj(n+1)},\phi_{sjn})\diamond\cdots\diamond(\psi_{sj2},\phi_{sj1})\diamond\hat{\theta}(\psi_{sj1})\big).\end{aligned} \tag{5.85}$$

where

$$\begin{aligned}\hat{\theta}(a)([\tau_1,\ldots,\tau_m,\bar{\tau}_1,\ldots,\bar{\tau}_l]_\bullet)&=\frac{1}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m)a(\bar{\tau}_1)\cdots a(\bar{\tau}_l),\\ \hat{\theta}(a)([\tau_1,\ldots,\tau_m,\bar{\tau}_1,\ldots,\bar{\tau}_l]_\blacktriangle)&=\frac{-2m}{m+1}\Phi(\tau_1)\cdots\Phi(\tau_m)a(\bar{\tau}_1)\cdots a(\bar{\tau}_l).\end{aligned} \tag{5.86}$$

The scheme is of order p if and only if

$$\Phi(\tau)=\xi(\tau)\quad \textit{for } |\tau|\le p, \tag{5.87}$$

where

$$\xi(\tau)=\begin{cases}\frac{1}{\gamma(\tau)} & \textit{if } \tau\in TP,\\ 0 & \textit{otherwise.}\end{cases} \tag{5.88}$$

As in section 5.4.2, we cut the branches between black and white nodes, regardless of the shape of the nodes, and denote this tree by τ^b. Number the nodes and reattach the cut-off parts. For the node i and the corresponding stem s^i, there exists a unique set of forests $\hat{\tau}^i = \{(\mu_1^i, \eta_1^i), \dots, (\mu_{n+1}^i, \eta_{n+1}^i)\}$ such that

$$\tau = [(\mu_1^i, \eta_1^i)]_{s_1^i} \circ \cdots [(\mu_n^i, \eta_n^i)]_{s_n^i} \circ [(\mu_{n+1}^i, \eta_{n+1}^i)]_{s_{n+1}^i}$$

Proposition 5.7. *The* Φ *of* (5.85) *satisfies*

$$\Phi(\tau) = \hat{e}(\tau) + \sum_{i=1}^{|\tau^b|} \Lambda(\hat{\tau}^i, s^i) \tag{5.89}$$

where $\hat{e}(\emptyset) = 1$ *and* $\hat{e}(\tau) = 0$ *for all* $\tau \neq \emptyset$, *and*

$$\begin{aligned}\Lambda(\hat{\tau}^i, s^i) = \theta([\mu_{n+1}^i]_{s_{n+1}^i})\Big(\sum_j b_{s^i j} \psi_{s^i j1}(\eta_1^i) \phi_{s^i j1}(\mu_1^i) \cdots \phi_{s^i jn}(\mu_n^i) \psi_{s^i j(n+1)}(\eta_{n+1}^i) \\ + (-1)^{|s^i|_\bullet - 1} \sum_j b_{\hat{s}^i j} \psi_{\hat{s}^i j(n+1)}(\eta_1^i) \phi_{\hat{s}^i jn}(\mu_1^i) \cdots \phi_{\hat{s}^i j1}(\mu_n^i) \psi_{\hat{s}^i j1}(\eta_n^i)\Big),\end{aligned} \tag{5.90}$$

with θ *given by* (5.74) *and* $\hat{s}^i$ *given by* $\hat{s}_k^i = s_{n-k+1}^i$ *for* $k = 1, \dots, n$, *and* $\hat{s}_{n+1}^i = s_{n+1}^i$.

The number of trees in TV grows very quickly. However, in our task of finding higher order schemes we may use the lessons of the previous chapters, and require that the arguments of S, $\nabla^2 H$ and Q in (5.84) are B-series up to order $p-1$. Then we only need to find order conditions for energy-preserving linear combinations of the form

$$\begin{aligned}\omega = [(\mu_1, \eta_1)]_{s_1} \circ \cdots \circ [(\mu_n, \eta_n)]_{s_n} \circ [\eta_{n+1}]_\bullet \\ + (-1)^n [(\mu_n, \eta_{n+1})]_{s_n} \circ \cdots \circ [(\mu_1, \eta_2)]_{s_1} \circ [\eta_1]_\bullet,\end{aligned} \tag{5.91}$$

where μ_i and η_i are forests of trees in $TP_\bullet$ and $TP_\circ$ respectively, for $i = 1, \dots, n+1$. Thus we can disregard any tree with $\triangle$ in it. Furthermore, we may color all nodes of the trees in μ_i and η_i except the roots gray, and let the elementary differentials corresponding to these trees be the same as the elementary differentials of B-trees.

We find the order conditions

$$\sum_{i \in I_l} \Lambda(\hat{\tau}^i, s^i) = \xi(\tau) - \hat{e}(\tau) - \sum_{i \in I_n} \Lambda(\hat{\tau}^i, s^i),$$

by using the relation

$$\Lambda(\{(\mu_1^i, \eta_1^i), \dots, (\mu_{n+1}^i, \eta_{n+1}^i)\}, s^i) = \hat{\theta}([\mu_{n+1}^i]_{s_{n+1}^i}) \Lambda(\{(\mu_1^i, \eta_1^i), \dots, (\emptyset, \eta_{n+1}^i)\}, \bar{s}^i)$$

to calculate $\Lambda(\hat{\tau}^i)$ for $i \in I_n$. The $\hat{\theta}$ is given by (5.81), and $\bar{s}^i$ is s^i with s_{n+1}^i replaced by $\bullet$.

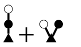
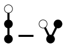
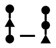
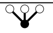
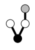
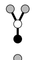
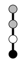
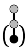
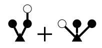
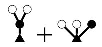
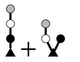
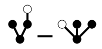
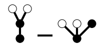
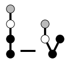
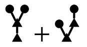
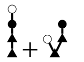
$\lvert\tau\rvert$	ω	s	Order condition
1	•	•	$\sum_j b_{sj} = \frac{1}{2}$
2		•	$2\sum_j b_{sj}\psi_{sj1}(\bullet) = \frac{1}{2}$
			$\sum_j b_{sj} = \frac{1}{2}$
3		•	$2\sum_j b_{sj}\psi_{sj1}(\bullet)^2 = \frac{1}{3}$
		•	$2\sum_j b_{sj}\psi_{sj1}($ $) = \frac{1}{6}$
			$\sum_j b_{sj}\phi_{sj1}(\bullet) = \frac{1}{3}$
			$\sum_j b_{sj}(\psi_{sj1}+\psi_{sj2})(\bullet) = \frac{1}{2}$
			$\sum_j b_{sj}(\psi_{sj2}-\psi_{sj1})(\bullet) = -\frac{1}{12}$
			$\sum_j b_{sj} = \frac{1}{2}$
			$\sum_j b_{sj} - \sum_j b_{\bar{s}j} = 0$
			$\sum_j b_{sj} = -\frac{1}{24}$
4		•	$2\sum_j b_{sj}\psi_{sj1}(\bullet)^3 = \frac{1}{4}$
		•	$2\sum_j b_{sj}\psi_{sj1}(\bullet)\psi_{0j1}($ $) = \frac{1}{8}$
		•	$2\sum_j b_{sj}\psi_{sj1}($ $) = \frac{1}{12}$
		•	$2\sum_j b_{sj}\psi_{sj1}($$) = \frac{1}{24}$
			$\sum_j b_{sj}\phi_{sj1}(\bullet)^2 = \frac{1}{4}$
			$\sum_j b_{sj}\phi_{sj1}($ $) = \frac{1}{6}$
			$\sum_j b_{sj}\psi_{sj1}(\bullet)\psi_{sj2}(\bullet) = \frac{1}{8}$
			$\sum_j b_{sj}\phi_{sj1}(\bullet)(\psi_{sj1}+\psi_{sj2})(\bullet) = \frac{1}{3}$
			$\sum_j b_{sj}(\psi_{sj1}(\bullet)^2+\psi_{sj2}(\bullet)^2) = \frac{1}{3}$
			$\sum_j b_{sj}(\psi_{sj1}+\psi_{sj2})($ $) = \frac{1}{6}$
			$\sum_j b_{sj}\phi_{sj1}(\bullet)(\psi_{sj2}-\psi_{sj1})(\bullet) = -\frac{1}{24}$
			$\sum_j b_{sj}(\psi_{sj2}(\bullet)^2-\psi_{sj1}(\bullet)^2) = -\frac{1}{12}$
			$\sum_j b_{sj}(\psi_{sj2}-\psi_{sj1})($ $) = -\frac{1}{24}$
			$\sum_{j,k} b_{sj}\phi_{sjk}(\bullet) = \frac{2}{3}$
			$\sum_j b_{sj}\psi_{sj2}(\bullet) = \frac{2}{3}$
			$\sum_j b_{sj}(\psi_{sj1}+\psi_{sj3})(\bullet) = \frac{1}{2}$

Table 5.6: Energy-preserving linear combinations of the form (5.91) and their associated order conditions for the discrete gradient method (5.9) with $\overline{S}(x,\hat{x},h)$ given by (5.84). Continued in Table 5.7.

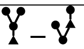
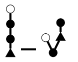
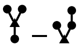
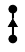
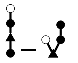
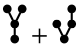
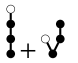
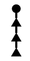
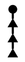
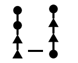
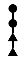
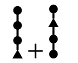
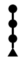
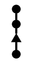
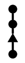
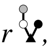
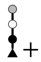
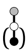
$\lvert\tau\rvert$	ω	s	Order condition
4			$\sum_j b_{sj}\phi_{sj2}(\bullet)-\sum_j b_{\bar{s}j}\phi_{\bar{s}j1}(\bullet)=0$
			$\sum_j b_{sj}(\psi_{sj1}-\psi_{sj3})(\bullet)=\frac{1}{12}$
			$\sum_j b_{sj}\phi_{sj2}(\bullet)-\sum_j b_{\bar{s}j}\phi_{\bar{s}j1}(\bullet)=0$
			$\sum_j b_{sj}\psi_{sj3}(\bullet)-\sum_j b_{\bar{s}j}\psi_{\bar{s}j1}(\bullet)=0$
			$\sum_{j,k} b_{sj}\phi_{sjk}(\bullet)=-\frac{1}{24}$
			$2\sum_j b_{2j}\psi_{2j1}(\bullet)=-\frac{1}{24}$
			$\sum_j b_{sj}(\psi_{sj1}+\psi_{sj3})(\bullet)=-\frac{1}{24}$
			$\sum_j b_{sj}=\frac{1}{2}$
			$\sum_j b_{sj}-\sum_j b_{\bar{s}j}=0$
			$\sum_j b_{sj}=-\frac{1}{12}$
			$\sum_j b_{sj}=0$

Table 5.7: Energy-preserving linear combinations of the form (5.91) and their associated order conditions for the discrete gradient method (5.9) with $\overline{S}(x,\hat{x},h)$ given by (5.84). Continuing from Table 5.6.

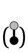
Example 5.7. *Consider* *, which is part of the energy-preserving linear combination*  $+$ *. We have two black nodes, and calculate*

$\Lambda(\hat{\tau}^1,s^1)=\Lambda(\{(\bullet,$ $)\},\blacktriangle)=\hat{\theta}($ $)\Lambda(\{(\varnothing,$ $)\},\bullet)=-\xi(\bullet)\xi($  $)=-\frac{1}{6},$

$\Lambda(\hat{\tau}^2,s^2)=\Lambda(\{(\varnothing,$ $),(\varnothing,\varnothing)\},$ $)=\sum_j b_{s^2j}\psi_{s^2j1}($ $)+\sum_j b_{\bar{s}^2j}\psi_{\bar{s}^2j2}($ $)$

$=\sum_j b_{s^2j}(\psi_{s^2j1}+\psi_{s^2j2})($ $).$

Hence the order condition associated to this linear combination is

$\sum_j b_{s^2j}(\psi_{s^2j1}+\psi_{s^2j2})($  $)=\frac{1}{6}.$

We consider the order conditions for trees with $|\tau| \leq 3$ displayed in Table 5.6, and find that

$$\begin{aligned}\overline{S}(x,\cdot,h) = &\frac{1}{4}S(x) + \frac{3}{4}S(z_3) + hS(z_2)Q(x,z_3)S(z_2)\\ &+\frac{1}{4}h\big(S(z_1)\nabla^2H(x)S(x) - S(x)\nabla^2H(x)S(z_1)\big)\\ &+h^2S(x)Q(x,x)S(x)Q(x,x)S(x)\\ &-\frac{1}{12}h^2\,S(x)\nabla^2H(x)S(x)\nabla^2H(x)S(x),\end{aligned} \tag{5.92}$$

where

$$z_1 = x + \frac{1}{3}hf(x), \qquad z_2 = x + \frac{1}{2}hf(x), \qquad z_3 = x + \frac{2}{3}hf(z_1),$$

guarantees third order convergence of the scheme (5.9) if $\overline{\nabla}H(x) \in C^2(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$. An approximation of $S(x)$ satisfying all the order conditions in tables 5.6 and 5.7 is given by

$$\begin{aligned}\overline{S}(x,\cdot,h) = &\frac{1}{2}(S(z_{11}+z_{12}) + S(z_{11}-z_{12}))\\ &+\frac{1}{12}h\big(S(z_6)\nabla^2H(z_2)S(x) - S(x)\nabla^2H(z_2)S(z_6)\big)\\ &+\frac{3}{7}h\big(S(z_3)Q(x,z_5)S(z_4) + S(z_4)Q(x,z_5)S(x,z_3)\big)\\ &+\frac{8}{105}hS(x)Q(x,z_7)S(x) + \frac{1}{15}hS(x)Q(x,x)S(x)\\ &+h^2\,S(z_2)Q(x,z_5)S(z_8)Q(x,z_5)S(z_2)\\ &-\frac{1}{12}h^2\,S(z_2)\nabla^2H(z_2)S(z_2)\nabla^2H(z_2)S(z_2)\\ &+\frac{1}{6}h^2(S(z_2)-S(x))\nabla^2H(x)S(x)Q(x,x)S(x)\\ &-\frac{1}{6}h^2S(x)Q(x,x)S(x)\nabla^2H(x)(S(z_2)-S(x))\\ &+h^3S(x)Q(x,x)S(x)Q(x,x)S(x)Q(x,x)\\ &-\frac{1}{12}h^3S(x)\nabla^2H(x)S(x)\nabla^2H(x)S(x)Q(x,x)S(x)\\ &-\frac{1}{12}h^3S(x)Q(x,x)S(x)\nabla^2H(x)S(x)\nabla^2H(x)S(x),\end{aligned} \tag{5.93}$$

with

$$z_1 = x + \frac{1}{3}hf(x), \quad z_2 = x + \frac{1}{2}hf(x), \quad z_3 = x + \frac{7-\sqrt{7}}{12}hf(z_1),$$
$$z_4 = x + \frac{7+\sqrt{7}}{12}hf(z_1), \quad z_5 = x + \frac{2}{3}hf(z_2), \quad z_6 = x + hf(z_2),$$
$$z_7 = x + \frac{5}{4}hf(z_2), \quad z_8 = x + \frac{4}{3}hf(z_2), \quad z_9 = x + hf(z_6), \quad z_{10} = x + hf(z_9),$$
$$z_{11} = \frac{1}{3}(x + z_2 + z_6) + \frac{1}{12}(-z_9 + z_{10}), \quad z_{12} = \frac{\sqrt{3}}{36}(7x - 2z_2 - 4z_6 + z_9 - 2z_{10}),$$

and hence a discrete gradient scheme with this $\overline{S}(x,\cdot,h)$ and any $\overline{\nabla}H(x,y) \in C^3(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$ will be of fourth order.

One advantage of choosing the AVF discrete gradient is that the resulting scheme generally requires fewer computations at each time step. This is clearly evident in the above example: if $\overline{\nabla} = \overline{\nabla}_{\mathrm{AVF}}$, then (5.92) collapses to (5.68), and (5.93) collapses to (5.70). However, if the AVF discrete gradient is difficult to calculate, there can also be much to gain in computational cost by choosing a symmetric discrete gradient, like the symmetrized Itoh–Abe discrete gradient (5.24) or the Furihata discrete gradient (5.26). Then one can ignore the order condition for any combination (5.91) for which $s_j = \blacktriangle$ and $\mu_j = \emptyset$ for some $j \in [1,n]$, since this corresponds to elementary differentials involving $Q(x,x)$, which we recall is zero when the discrete gradient is symmetric. If we consider the conditions for fourth order presented in tables 5.6 and 5.7, this eliminates 17 of the 22 conditions for trees with $\blacktriangle$ in the stem. By considering the remaining order conditions we get that, if $\overline{\nabla}H(x,y) \in C^3(\mathbb{R}^d \times \mathbb{R}^d, \mathbb{R}^d)$ and $\overline{\nabla}H(x,y) = \overline{\nabla}H(y,x)$, the discrete gradient scheme (5.9) is of fourth order if

$$\begin{aligned} \overline{S}(x,\cdot,h) = &\frac{1}{2}(S(z_5+z_6) + S(z_5-z_6)) + \frac{8}{9}hS(z_1)Q(z_7)S(z_1) \\ &+ \frac{1}{12}h\left(S(z_2)\nabla^2 H(z_1)S(x) - S(x)\nabla^2 H(z_1)S(z_2)\right) \\ &- \frac{1}{12}h^2 S(z_1)\nabla^2 H(z_1)S(z_1)\nabla^2 H(z_1)S(z_1), \end{aligned} \tag{5.94}$$

with

$$z_1 = x + \frac{1}{2}hf(x), \qquad z_2 = x + hf(z_1), \qquad z_3 = x + hf(z_2),$$
$$z_4 = x + hf(z_3), \qquad z_5 = \frac{1}{3}(x + z_1 + z_2) + \frac{1}{12}(-z_3 + z_4),$$
$$z_6 = \frac{\sqrt{3}}{36}(7x - 2z_1 - 4z_2 + z_3 - 2z_4), \qquad z_7 = x + \frac{3}{4}hf(z_1).$$

If S is constant, (5.94) simplifies to

$$\overline{S}(x,\cdot,h) = S + \frac{8}{9}hSQ(z_7)S - \frac{1}{12}h^2 S\nabla^2 H(z_1)S\nabla^2 H(z_1)S. \tag{5.95}$$

5.6 Numerical experiments and conclusions

The Hénon–Heiles system can be written on the form (5.6) with

$$S = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}, \qquad H(q,p) = \frac{1}{2}(q_1^2 + q_2^2 + p_1^2 + p_2^2) + q_1^2 q_2 - \frac{1}{3} q_2^3, \tag{5.96}$$

where I is the 2×2 identity matrix. We use here the same initial conditions used in [26]: $q_1 = \frac{1}{10}$, $q_2 = -\frac{1}{2}$, $p_1 = p_2 = 0$. The order of some of the energy-preserving methods proposed in this paper are confirmed by the left plot in Figure 5.1. We compare the performance of the fourth order discrete gradient methods obtained by using the $\overline{S}$ given by (5.29) coupled with three different discrete gradients: the Itoh–Abe discrete gradient (5.23), the Furihata discrete gradient (5.26), and the AVF discrete gradient (5.10). The symmetrized Itoh–Abe discrete gradient (5.24) is for this H identical to the Furihata discrete gradient. The AVF and Furihata discrete gradient methods perform in this case very similarly, and thus the error from the Furihata discrete gradient method is excluded from the right plot in Figure 5.1. We observe that, although it initially performs on par with the AVF method, the Itoh–Abe discrete gradient method gives a lower global error than the other fourth order methods as time goes on. Note however that this method requires the most computations at every time step.

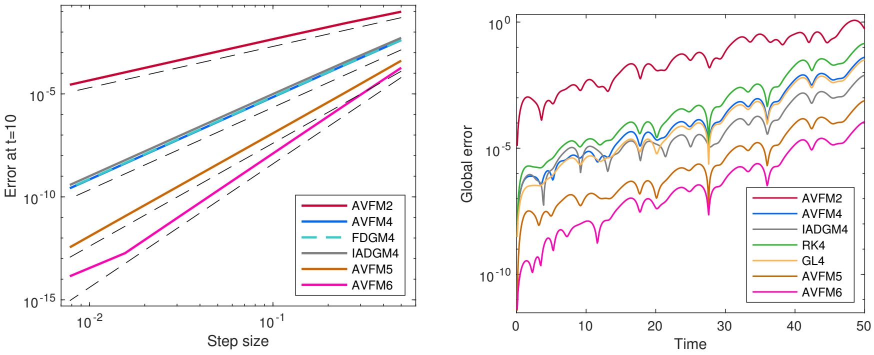


Figure 5.1: Error plots for the Hénon–Heiles system (5.96) solved by various discrete gradient methods: AVFM2 is the standard AVF method (5.11); AVFM4 is the AVF discrete gradient method with $\overline{S}$ given by (5.30); FDGM4 is the Furihata discrete gradient method with $\overline{S}$ given by (5.95); IADGM4 is the Itoh–Abe discrete gradient method with $\overline{S}$ given by (5.29); AVFM5 is the scheme (5.49); AVFM6 is (5.51). RK4 is the classic Runge–Kutta method and GL4 is the fourth order Gauss–Legendre method, included for comparison. The black dashed lines in the order plot are reference lines of order two, four, five and six. The step size in the plot to the right is $h = 0.1$.

The methods should also be tested on a skew-gradient system with non-constant S. We choose the Lotka–Volterra system also used for numerical experiments in [5]. It is given by

$$S = \frac{1}{2}\begin{pmatrix} 0 & -x_1x_2 & x_1x_3 \\ x_1x_2 & 0 & -2x_2x_3 \\ -x_1x_3 & 2x_2x_3 & 0 \end{pmatrix}, \qquad H(x) = 2x_1 + x_2 + 2x_3 + \ln(x_2) - 2\ln(x_3),$$

and initial conditions $x_1 = 1$, $x_2 = \frac{19}{10}$, $x_3 = \frac{1}{2}$. For this H, the Itoh–Abe, Furihata and AVF discrete gradients are all equivalent. We consider fourth order discrete gradient methods where ∇S is given either dependent on or independent of $\hat{x}$; that is, (5.69) or (5.70). The implicitly given (5.70) yields a significantly lower error in the solution of the corresponding discrete gradient method, as can be witnessed from the left plot in Figure 5.2. In contrast to what we observed for the canonical Hamiltonian system studied above, none of the discrete gradient methods give a global error lower than that of the fourth order Gauss–Legendre method.

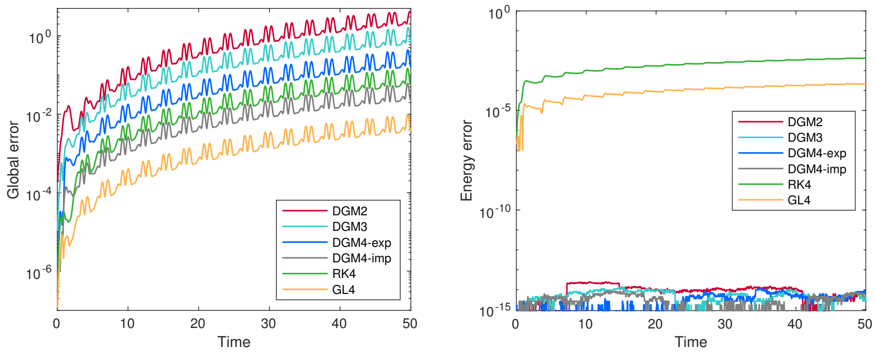


Figure 5.2: Error in the solution and in the energy for discrete gradient methods with $\overline{S}$ given by $\overline{S}(x,\hat{x}) = S(\frac{x+\hat{x}}{2})$ for DGM2, (5.68) for DGM3, (5.70) for DGM4-exp and (5.69) for DGM4-imp, applied to the Lotka–Volterra system, with step size $h = 0.05$. For comparison, errors from using the standard fourth order Runge–Kutta (RK4) and Gauss–Legendre (GL4) methods are also included.

The main purpose of this paper has been to develop order theory for discrete gradient methods, rather than the development of specific schemes. Hence we have simply proposed some higher order schemes satisfying the derived order conditions; analysis to find more optimal schemes is something we leave for the future. After such an analysis is performed, the methods could be tested on more advanced problems than those considered above, e.g. for the temporal discretization of Hamiltonian partial differential equations, and their performance as measured by accuracy relative to computational cost could be compared to existing methods.

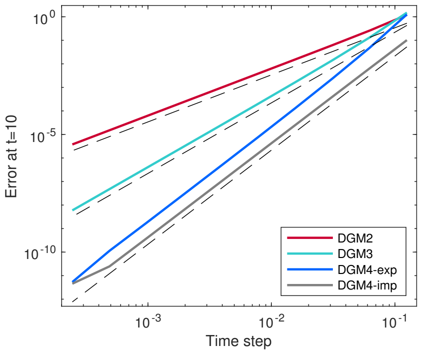


Figure 5.3: Order or discrete gradient methods applied to the Lotka–Volterra system, with different $\overline{S}$: $\overline{S}(x,\hat{x}) = S(\frac{x+\hat{x}}{2})$ for DGM2, (5.68) for DGM3, (5.70) for DGM4-exp, (5.69) for DGM4-imp. The dashed lines are reference lines of order two, three and four.

The order theory presented here can possibly be developed further in a couple of different directions. The schemes given in this paper with $\overline{S}$ independent of $\hat{x}$ are linearly implicit when H is quadratic; if the order theory is extended to the polarized discrete gradient methods of [6, 18], we could get higher order linearly implicit multi-step schemes for systems with polynomial first integrals of any degree. Another avenue could be to consider order conditions for the discrete Riemannian gradient methods presented in [1]. Then the results in the previous chapter are especially interesting, since the integral in the AVF discrete Riemannian gradient can be challenging to compute analytically.

Acknowledgments

The author is grateful to Brynjulf Owren for the invaluable suggestions he has provided during the writing of this paper. This research was supported by the Research Council of Norway, project no. 231632. Some of the work was undertaken while the author was a guest at the Isaac Newton Institute for Mathematical Sciences in Cambridge.

Bibliography

[1] E. Celledoni, S. Eidnes, B. Owren, and T. Ringholm. Energy-preserving methods on Riemannian manifolds. *Math. Comp.*, 89(322):699–716, 2020.

[2] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, G. R. W. Quispel, and W. M. Wright. Energy-preserving Runge-Kutta methods. *M2AN Math. Model. Numer. Anal.*, 43(4):645–649, 2009.

[3] P. Chartier, E. Faou, and A. Murua. An algebraic approach to invariant preserving integrators: the case of quadratic and Hamiltonian invariants. *Numer. Math.*, 103(4):575–590, 2006.

[4] A. J. Chorin, M. F. McCracken, T. J. R. Hughes, and J. E. Marsden. Product formulas and numerical algorithms. *Comm. Pure Appl. Math.*, 31(2):205–256, 1978.

[5] D. Cohen and E. Hairer. Linear energy-preserving integrators for Poisson systems. *BIT*, 51(1):91–101, 2011.

[6] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33(5):2318–2340, 2011.

[7] S. Eidnes, B. Owren, and T. Ringholm. Adaptive energy preserving methods for partial differential equations. *Adv. Comput. Math.*, 44(3):815–839, 2018.

[8] E. Faou, E. Hairer, and T.-L. Pham. Energy conservation with non-symplectic methods: examples and counter-examples. *BIT*, 44(4):699–709, 2004.

[9] D. Furihata. Finite difference schemes for $\partial u/\partial t = (\partial/\partial x)^{\alpha}\delta G/\delta u$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156(1):181–205, 1999.

[10] D. Furihata and T. Matsuo. *Discrete variational derivative method*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011. A structure-preserving numerical method for partial differential equations.

[11] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[12] E. Hairer. Energy-preserving variant of collocation methods. *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, 5(1-2):73–84, 2010.

[13] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[14] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, 25(1):35–61, 1983.

[15] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 76(1):85–102, 1988.

[16] Y. A. Kriksin. A conservative difference scheme for a system of Hamiltonian equations with external action. *Zh. Vychisl. Mat. i Mat. Fiz.*, 33(2):206–218, 1993.

[17] J. E. Marsden. *Lectures on mechanics*, volume 174 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 1992.

[18] T. Matsuo and D. Furihata. Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations. *J. Comput. Phys.*, 171(2):425–447, 2001.

[19] T. Matsuo, M. Sugihara, D. Furihata, and M. Mori. Spatially accurate dissipative or conservative finite difference schemes derived by the discrete variational method. *Japan J. Indust. Appl. Math.*, 19(3):311–330, 2002.

[20] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci.*, 357(1754):1021–1045, 1999.

[21] D. I. McLaren and G. R. W. Quispel. Integral-preserving integrators. *J. Phys. A*, 37(39):L489–L495, 2004.

[22] D. I. McLaren and G. R. W. Quispel. Bootstrapping discrete-gradient integral-preserving integrators to fourth order. In M. Daniel and S. Rajasekar, editors, *Nonlinear dynamics*, pages 157–171. Narosa Publishing House, 2009.

[23] S. P. Nørsett and A. Wolfbrandt. Order conditions for Rosenbrock type methods. *Numer. Math.*, 32(1):1–15, 1979.

[24] R. A. Norton, D. I. McLaren, G. R. W. Quispel, A. Stern, and A. Zanna. Projection methods and discrete gradient methods for preserving first integrals of ODEs. *Discrete Contin. Dyn. Syst.*, 35(5):2079–2098, 2015.

[25] R. A. Norton and G. R. W. Quispel. Discrete gradient methods for preserving a first integral of an ordinary differential equation. *Discrete Contin. Dyn. Syst.*, 34(3):1147–1170, 2014.

[26] G. R. W. Quispel and D. I. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A*, 41(4):045206, 7, 2008.

[27] T. Yaguchi, T. Matsuo, and M. Sugihara. The discrete variational derivative method based on discrete differential forms. *J. Comput. Phys.*, 231(10):3963–3986, 2012.

Linearly implicit structure-preserving schemes for Hamiltonian systems

Sølve Eidnes, Lu Li and Shun Sato

To appear in Journal of Computational and Applied Mathematics

Linearly implicit structure-preserving schemes for Hamiltonian systems

Abstract. Kahan's method and a two-step generalisation of the discrete gradient method are both linearly implicit methods that can preserve a modified energy for Hamiltonian systems with a cubic Hamiltonian. These methods are here investigated and compared. The schemes are applied to the Korteweg–de Vries equation and the Camassa–Holm equation, and the numerical results are presented and analysed.

6.1 Introduction

The field of geometric numerical integration has garnered increased attention over the last three decades. It considers the design and analysis of numerical methods that can capture geometric properties of the flow of the differential equation to be modelled. These geometric properties are mainly invariants over time; they are conserved quantities such as Hamiltonian energy, angular momentum, volume or symplecticity. Among them the conservation of energy is particularly important for proving the existence and uniqueness of solutions for partial differential equations (PDEs) [19]. Numerical schemes inheriting such properties from the continuous dynamical system have been shown in many cases to be advantageous, especially when integration over long time intervals is considered [10].

For general non-linear differential equations, one may use a standard fully implicit scheme to solve a problem numerically. Then a non-linear system of equations must be solved at each time step. Typically this is done by the use of an iterative solver where a linear system is to be solved at each iteration. This quickly becomes a computationally expensive procedure, especially since the number of iterations needed in general increases with the size of the system; see a numerical example comparing the computational cost for implicit and linearly implicit methods in [5]. A fully explicit method on the other hand, may over-simplify the problem and lead to the loss of important information, and will often have inferior stability properties. The golden middle way may be found in linearly implicit schemes, i.e. schemes where the non-linear terms are discretized such that the solution at the next time step is found from solving one linear system.

Our aim is to present and analyse linearly implicit schemes with preservation properties. We consider ordinary differential equations (ODEs) that can be written in the form

$$\begin{aligned} \dot{x} = f(x) = S\nabla H(x), \quad x \in \mathbb{R}^d, \\ x(0) = x_0, \end{aligned} \tag{6.1}$$

where S is a constant skew-symmetric matrix and H is a cubic Hamiltonian function. The famous geometric characteristic for equations like (6.1) is that the exact flow is energy-preserving,

$$\frac{d}{dt}H(x) = \nabla H(x)^T \frac{dx}{dt} = \nabla H(x)^T S \nabla H(x) = 0,$$

and symplectic if S is the canonical skew-symmetric matrix:

$$\Psi_{y_0}(t)^T S \Psi_{y_0}(t) = S, \tag{6.2}$$

where $\Psi_{y_0}(t) := \frac{\partial \varphi_t(y_0)}{\partial y_0}$, with $\varphi_t : \mathbb{R}^d \to \mathbb{R}^d$, $\varphi_t(y_0) = y(t)$ the flow map of (6.1) [10]. A numerical one-step method is said to be energy-preserving if H is constant along the numerical solution, and symplectic if the numerical flow map is symplectic. Both the energy-preserving methods and the symplectic methods, the latter of which has the ability to preserve a perturbation of the Hamiltonian H of (6.1), have their own advantages. In particular, the energy-preserving property has been found to be crucial in the proof of stability for several such numerical methods, see e.g [6]. However, there is no numerical integration method that can be simultaneously symplectic and energy-preserving for general Hamiltonian systems [20]. In this paper we will focus on energy-preserving numerical integration.

We wish to study and compare two types of existing methods with geometric properties. The first one is Kahan's method for quadratic ODE vector fields [12], which by construction is linearly implicit, and for which the geometric properties have been studied in [3]. Kahan's method has not been extensively studied for solving PDEs so far, with the notable exception [13]. This is a one-step method, but we will also give its formulation as a two-step method in this paper, for easier comparison to the other method to be studied. That method, which we call the polarised discrete gradient (PDG) method, is based on the multiple points discrete variational derivative method for PDEs presented by Furihata, Matsuo and coauthors in the papers [14–16] and the monograph [8]. A more general framework for such schemes is given by Dahlby and Owren in [5]. With the aim of easing the comparison to Kahan's method, we present here the two-step method of [5, 8] as it looks for ODEs of the form (6.1). When Hamiltonian PDEs are considered, by semi-discretizing in space to obtain a system of Hamiltonian ODEs and then applying the PDG method, one may obtain the schemes of the aforementioned references; a specific scheme will depend on the choice of spatial discretization as well as the choices of some functions to be explained in the next section: the polarised energy and the polarised discrete gradient.

This paper is divided into two main parts. In the next chapter, we present the methods in consideration, and give some theoretical results on their geometric properties. In Chapter 3, we present numerical results for the Camassa–Holm equation and the Korteweg–de Vries equation, including analysis of stability and dispersion, comparing the methods.

6.2 Linearly implicit schemes

We will present an ODE formulation of the linearly implicit schemes presented by Furihata, Matsuo and coauthors in [8, 14–16] and by Dahlby and Owren in [5]. Inspired by the nomenclature of the latter reference, we call these schemes polarised discrete gradient methods. Then we present a special case of this polarisation method in the same framework as Kahan's method, with the goal of obtaining more clarity in comparison of the methods.

6.2.1 Polarised discrete gradient methods

The idea behind the PDG methods is to generalise the discrete gradient method in such a way that a relaxed variant of the preservation property is intact, while nonlinear terms are discretized over consecutive time steps to ensure linearity in the scheme. Let us first recall the concept of discrete gradient methods. A discrete gradient is a continuous map $\overline{\nabla} H : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ such that for any $x, y \in \mathbb{R}^d$

$$H(y) - H(x) = (y - x)^T \overline{\nabla} H(x, y).$$

The discrete gradient method for (6.1) is then given by

$$\frac{x^{n+1} - x^n}{\Delta t} = S \overline{\nabla} H(x^n, x^{n+1}),$$

which will preserve the energy of the system (6.1) at any time step. Here and in what follows, x^n is the numerical approximation of x at $t = t_n$ and x_k^n is the numerical approximation of the kth component of x at $t = t_n$.

Restricting ourselves to two-step methods, we define the PDG methods as follows.

Definition 6.1. For the energy H of (6.1), consider the polarised energy as a function $\tilde{H} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ satisfying the properties

$$\tilde{H}(x, x) = H(x),$$
$$\tilde{H}(x, y) = \tilde{H}(y, x).$$

A polarised discrete gradient (PDG) for $\tilde{H}$ is a function $\overline{\nabla}\tilde{H} : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ satisfying

$$\begin{aligned} \tilde{H}(y,z) - \tilde{H}(x,y) &= \frac{1}{2}(z-x)^T \overline{\nabla}\tilde{H}(x,y,z), \qquad (6.3)\\ \overline{\nabla}\tilde{H}(x,x,x) &= \nabla H(x), \end{aligned}$$

and the corresponding polarised discrete gradient scheme is given by

$$\frac{x^{n+2}-x^n}{2\Delta t} = S\overline{\nabla}\tilde{H}(x^n, x^{n+1}, x^{n+2}). \qquad (6.4)$$

Proposition 6.1. The numerical scheme (6.4) preserves the polarised invariant $\tilde{H}$ in the sense that $\tilde{H}(x^n, x^{n+1}) = \tilde{H}(x^0, x^1)$ for all $n \geq 0$.

Proof.

$$\begin{aligned} \tilde{H}(x^{n+1}, x^{n+2}) - \tilde{H}(x^n, x^{n+1}) &= \frac{1}{2}(x^{n+2} - x^n)^T \overline{\nabla}\tilde{H}(x^n, x^{n+1}, x^{n+2})\\ &= \Delta t \overline{\nabla}\tilde{H}(x^n, x^{n+1}, x^{n+2})^T S \overline{\nabla}\tilde{H}(x^n, x^{n+1}, x^{n+2})\\ &= 0, \end{aligned}$$

where the last equality follows from the skew-symmetry of S. □

We remark here that in the cases where we seek a time-stepping scheme for the system of Hamiltonian ODEs resulting from discretizing a Hamiltonian PDE in space in an appropriate manner, e.g. as described in [2], H will be a discrete approximation to an integral $\mathcal{H}$. Thus a two-step PDG method and a standard one-step discrete gradient method, the latter in general fully implicit, will preserve two different discrete approximations separately to the same $\mathcal{H}$.

The task of finding a PDG satisfying (6.3) is approached differently in our two main references, [8,14–16] and [5]. Furihata, Matsuo and coauthors apply a generalisation of the approach introduced by Furihata in [7] for finding discrete variational derivatives, while Dahlby and Owren suggest a generalisation of the average vector field (AVF) discrete gradient [17], given by

$$\overline{\nabla}_{\text{AVF}}\tilde{H}(x,y,z) = 2\int_0^1 \nabla_x \tilde{H}(\xi x + (1-\xi)z, y)\, \mathrm{d}\xi,$$

where $\nabla_x \tilde{H}(x,y)$ is the gradient of $\tilde{H}(x,y)$ with respect to its first argument. Provided that the spatial discretization is performed in the same way, these two approaches lead to the same scheme for an $\tilde{H}$ quadratic in each of its arguments, as does a generalisation of the midpoint discrete gradient of Gonzalez [9]. Based on this, we present the most straightforward approach for finding this specific PDG for the cases we are studying in this paper:

Proposition 6.2. Given an $\tilde{H}(x,y)$ that is at most quadratic in each of its arguments, define $\nabla_x\tilde{H}(x,y)$ as the gradient of $\tilde{H}$ with respect to its first argument. Then a PDG for $\tilde{H}$ is given by

$$\overline{\nabla}\tilde{H}(x,y,z) = 2\nabla_x\tilde{H}(\frac{x+z}{2},y). \tag{6.5}$$

Proof. We may write

$$\tilde{H}(x,y) = x^T A(y)x + b(y)^T x + c(y),$$

for some symmetric $A:\mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$, $b:\mathbb{R}^d \to \mathbb{R}^d$ and $c:\mathbb{R}^d \to \mathbb{R}$. Then

$$\nabla_x\tilde{H}(x,y) = 2A(y)x + b(y),$$

and

$$\begin{aligned}\nabla_x\tilde{H}(\frac{x+z}{2},y)^T(z-x) &= (2A(y)\frac{x+z}{2} + b(y))^T(z-x)\\ &= z^T A(y)z + b(y)^T z - x^T A(y)x - b(y)^T x\\ &= \tilde{H}(y,z) - \tilde{H}(x,y).\end{aligned}$$

Furthermore,

$$\overline{\nabla}\tilde{H}(x,x,x) = 2\nabla_x\tilde{H}(x,x) = \nabla H(x).$$

□

As remarked in Theorem 4.5 of [5]: if the polarised energy $\tilde{H}(x,y)$ is at most quadratic in each of its arguments, the scheme (6.4) with the PDG (6.5) is linearly implicit.

An alternative to (6.5) could be a generalisation of the Itoh–Abe discrete gradient [11], defined by its i-th component

$$\overline{\nabla}_{\text{IA}}\tilde{H}(x,y,z)_i = 2\begin{cases}\bar{\partial}\tilde{H}(x,y,z)_i & \text{if } x_i \neq z_i,\\ \frac{\partial\tilde{H}}{\partial x_i}((z_1,\dots,z_{i-1},x_i,\dots,x_d),y) & \text{if } x_i = z_i,\end{cases}$$

where

$$\bar{\partial}\tilde{H}(x,y,z)_i = \frac{\tilde{H}((z_1,\dots,z_i,x_{i+1},\dots,x_d),y) - \tilde{H}((z_1,\dots,z_{i-1},x_i,\dots,x_d),y)}{z_i - x_i}.$$

A symmetrized variant of this, given by $\overline{\nabla}_{\text{SIA}}\tilde{H}(x,y,z) := \frac{1}{2}(\overline{\nabla}_{\text{IA}}\tilde{H}(x,y,z) + \overline{\nabla}_{\text{IA}}\tilde{H}(z,y,x))$ is again identical to (6.5), whenever $\tilde{H}$ is quadratic in each of its arguments.

6.2.2 A general framework and Kahan's method

For ODEs of the form (6.1), consider the two-step schemes of the form

$$\frac{x^{n+2}-x^n}{2\Delta t} = S\sum_{i,j=1}^{3} \alpha_{ij}(H''(x^{n-1+i})x^{n-1+j} + \beta(x^{n-1+i})), \tag{6.6}$$

where $H'' : \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$ is the Hessian matrix of H and $\beta(x) := 2\nabla H(x) - H''(x)x$. For cubic H, this scheme is linearly implicit if and only if $\alpha_{33} = 0$.

In this section, we first consider the case when the Hamiltonian is a cubic homogeneous polynomial, in which case the term $\beta(x)$ in (6.6) will disappear, and then generalise the results to the non-homogeneous case.

Theorem 6.3. *The scheme* (6.6) *with* $\alpha_{21} = \alpha_{23} = \frac{1}{4}$, $\alpha_{ij} = 0$ *otherwise, i.e.*

$$\frac{x^{n+2}-x^n}{2\Delta t} = \frac{1}{4}SH''(x^{n+1})(x^n + x^{n+2}), \tag{6.7}$$

where x^1 *is found from* x^0 *by Kahan's method, is equivalent to Kahan's method over two consecutive steps, when applied to ODEs of the form* (6.1) *with homogeneous cubic* H.

Proof. As shown in [3], Kahan's method can be written into a Runge–Kutta form

$$\frac{x^{n+1}-x^n}{\Delta t} = -\frac{1}{2}f(x^n) + 2f(\frac{x^n + x^{n+1}}{2}) - \frac{1}{2}f(x^{n+1}). \tag{6.8}$$

Two steps of this can be written as

$$\begin{aligned}\frac{x^{n+2}-x^n}{2\Delta t} =& -\frac{1}{4}f(x^n) - \frac{1}{2}f(x^{n+1}) - \frac{1}{4}f(x^{n+2}) \\ &+ f(\frac{x^n + x^{n+1}}{2}) + f(\frac{x^{n+1} + x^{n+2}}{2}).\end{aligned} \tag{6.9}$$

Using that for a homogeneous cubic H we have $\nabla H(x) = \frac{1}{2}H''(x)x$, $H''(x)y = H''(y)x$ and $H''(x+y) = H''(x) + H''(y)$, and inserting $f(x) = S\nabla H(x)$ in (6.9), we get (6.7). On the other hand, if we have found x^{n+1} by Kahan's method and x^{n+2} by (6.9), we see that by subtracting (6.8) from (6.9) we get (6.8) with n replaced by $n+1$. □

Remark 6.1. *The scheme* (6.7) *with the first step computed by Kahan's method preserves the polarised invariant* $\tilde{H}(x^n, x^{n+1}) = \frac{1}{6}(x^n)^T H''(\frac{x^n+x^{n+1}}{2})x^{n+1}$*, since Kahan's method preserves this polarised invariant [3]. We note that the scheme* (6.7) *satisfies*

$$(x^n)^T H''(x^n)x^{n+1} = (x^{n+1})^T H''(x^{n+2})x^{n+2},$$

independent of how x^1 is found, following from the skew symmetry of the matrix S. However, it preserves the polarised invariant $\frac{1}{6}(x^n)^T H''(\frac{x^n+x^{n+1}}{2})x^{n+1}$ only if Kahan's method or an equivalent scheme is used to calculate x^1 from x^0.

A special case of the PDG method which preserves the same polarised Hamiltonian as Kahan's method, can also be written on the form (6.6):

Theorem 6.4. *For a homogeneous cubic H and the polarised energy given by*

$$\tilde{H}(x,y) = \frac{1}{6}x^T H''(\frac{x+y}{2})y,$$

the scheme (6.4) *with the PDG* (6.5) *applied to* (6.1) *is equivalent to* (6.6) *with $\alpha_{21} = \alpha_{22} = \alpha_{23} = \frac{1}{6}$, $\alpha_{ij} = 0$ otherwise, i.e.*

$$\frac{x^{n+2} - x^n}{2\Delta t} = \frac{1}{6}SH''(x^{n+1})(x^n + x^{n+1} + x^{n+2}). \tag{6.10}$$

Proof.

$$\nabla_x \tilde{H}(x,y) = \frac{1}{6}H''(\frac{x+y}{2})y + \frac{1}{6}H''(\frac{y}{2})x = \frac{1}{12}H''(2x+y)y,$$

and thus

$$\overline{\nabla}\tilde{H}(x,y,z) = 2\nabla_x \tilde{H}(\frac{x+z}{2}, y) = \frac{1}{6}H''(x+y+z)y = \frac{1}{6}H''(y)(x+y+z).$$

□

It can be shown that many well known Runge–Kutta methods performed over two consecutive steps are methods in the class (6.6) when applied to (6.1) with H cubic. As two examples, the implicit midpoint method over two steps is (6.6) with $\alpha_{11} = \alpha_{33} = \frac{1}{16}, \alpha_{21} = \alpha_{22} = \alpha_{23} = \frac{1}{8}$, $\alpha_{ij} = 0$ otherwise, while the trapezoidal rule is (6.6) with $\alpha_{11} = \alpha_{33} = \frac{1}{8}, \alpha_{22} = \frac{1}{4}$, $\alpha_{ij} = 0$ otherwise. The integral-preserving average vector field method [18] over two steps is (6.6) with $\alpha_{11} = \alpha_{21} = \alpha_{23} = \alpha_{33} = \frac{1}{12}, \alpha_{22} = \frac{1}{6}$, $\alpha_{ij} = 0$ otherwise.

Now, in the cases where H is non-homogeneous, one can use the technique employed in [3], i.e. adding one variable x_0 to generate an equivalent problem to the original one, for a homogeneous Hamiltonian $\bar{H} : \mathbb{R}^{d+1} \to \mathbb{R}$ defined such that $\bar{H}(1, x_1, \ldots, x_d) = H(x_1, \ldots, x_d)$. Also constructing the $(d+1)\times(d+1)$ skew-symmetric matrix $\bar{S}$ by adding a zero initial row and a zero initial column to S, we get that solving the system

$$\begin{aligned} \dot{\bar{x}} &= \bar{S}\nabla\bar{H}(\bar{x}), \quad \bar{x} \in \mathbb{R}^{d+1} \\ \bar{x}(0) &= (1, x^0), \end{aligned} \tag{6.11}$$

is equivalent to solving (6.1). Following the above results for the homogeneous $\bar{H}$ and (6.11), we can generalise Theorem 6.3 and Theorem 6.4 for all cubic H. Generalisations of the preservation properties follow directly; e.g., Kahan's method and the PDG method can preserve the perturbed energy $\tilde{H}(x^n, x^{n+1}) := \frac{1}{6}(\bar{x}^n)^T \bar{H}''(\frac{\bar{x}^n+\bar{x}^{n+1}}{2})\bar{x}^{n+1}$ also for non-homogeneous cubic H.

6.3 Numerical experiments

To have a better understanding of the above methods, we will apply them to systems of two different PDEs: the Korteweg–de Vries (KdV) equation and the Camassa–Holm equation. We will compare our methods to the midpoint method, which is a symplectic, fully implicit method. We solve the two PDEs by discretizing in space to obtain a Hamiltonian ODE system of the type (6.1) and then applying the PDG method (denoted by PDGM), Kahan's method (Kahan) and the midpoint method (MP) to this.

6.3.1 Camassa–Holm equation

In this section, we consider the Camassa–Holm equation

$$u_t - u_{xxt} + 3uu_x = 2u_x u_{xx} + uu_{xxx}$$

defined on the periodic domain $\mathbb{S} := \mathbb{R}/L\mathbb{Z}$. It has the conserved quantities

$$\mathcal{H}_1[u] = \frac{1}{2}\int_{\mathbb{S}} (u^2 + u_x^2)\,\mathrm{d}x, \qquad \mathcal{H}_2[u] = \frac{1}{2}\int_{\mathbb{S}} \left(u^3 + uu_x^2\right)\mathrm{d}x.$$

Here we consider the variational form of the Hamiltonian $\mathcal{H}_2$:

$$(1-\partial_x^2)u_t = -\partial_x \frac{\delta \mathcal{H}_2}{\delta u}, \qquad \frac{\delta \mathcal{H}_2}{\delta u} = \frac{3}{2}u^2 + \frac{1}{2}u_x^2 - (uu_x)_x. \tag{6.12}$$

We follow the approach presented in [2] and semi-discretize the energy $\mathcal{H}_2$ of (6.12) as

$$H_2(u)\Delta x = \frac{1}{2}\sum_{k=1}^{K}\left(u_k^3 + u_k \frac{(\delta_x^+ u_k)^2 + (\delta_x^- u_k)^2}{2}\right)\Delta x, \tag{6.13}$$

where the difference operators δ_x^+ and δ_x^- are defined by

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

For later use, we here also introduce the notation

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

$$\mu_x^+ u_k := \frac{u_{k+1} + u_k}{2}, \qquad \mu_x^- u_k := \frac{u_k + u_{k-1}}{2},$$

and the matrices corresponding to the difference operators δ_x^+, δ_x^-, $\delta_x^{\langle 1\rangle}$, $\delta_x^{\langle 2\rangle}$, μ_x^+ and μ_x^-, which are denoted by D^+, D^-, $D^{\langle 1\rangle}$, $D^{\langle 2\rangle}$, M^+ and M^-. Denoting the numerical solution $U = [u_1, \dots, u_K]^T$, and by using the properties of the above difference operators, we thus get

$$\nabla H_2(U) = \frac{3}{2}U_{.}^{2} + \frac{1}{2}M^-(D^+U)_{.}^{2} - \frac{1}{2}D^{\langle 2\rangle}U_{.}^{2},$$

where $U_{.}^{2}$ is the elementwise square of U. Then the semi-discretized system for the Camassa–Holm equation becomes

$$\dot{U} = S\nabla H_2(U) = -(I - D^{\langle 2\rangle})^{-1}D^{\langle 1\rangle}\nabla H_2(U). \tag{6.14}$$

The above-mentioned schemes applied to (6.14) give us

$$(I - D^{\langle 2\rangle})\frac{U^{n+1} - U^n}{\Delta t} = -D^{\langle 1\rangle}\nabla H_2(\frac{U^{n+1} + U^n}{2}), \qquad \text{(MP)}$$

$$(I - D^{\langle 2\rangle})\frac{U^{n+1} - U^n}{\Delta t} = -\frac{1}{2}D^{\langle 1\rangle}H_2^{''}(U^n)U^{n+1}, \qquad \text{(Kahan)}$$

$$(I - D^{\langle 2\rangle})\frac{U^{n+2} - U^n}{2\Delta t} = -D^{\langle 1\rangle}\overline{\nabla}\tilde{H}_2(U^n, U^{n+1}, U^{n+2}), \qquad \text{(PDGM)}$$

where $H_2^{''}(U) = 3\,\text{diag}(U) + M^-\,\text{diag}(D^+U)D^+ - D^{\langle 2\rangle}\,\text{diag}(U)$ is the Hessian of $H_2(U)$ and
$\overline{\nabla}\tilde{H}_2(U^n, U^{n+1}, U^{n+2})$ is the PDG of Proposition 6.2 with polarised discrete energy

$$\begin{aligned}\tilde{H}_2(U^n, U^{n+1})\Delta x := \frac{1}{2}\sum_{k=1}^{K}\Bigg(& u_k^n u_k^{n+1}\frac{u_k^n + u_k^{n+1}}{2} \\ &+ a(\mu_x^+\frac{u_k^n + u_k^{n+1}}{2})(\delta_x^+ u_k^n)(\delta_x^+ u_k^{n+1}) \\ &+ (1-a)\frac{(\mu_x^+ u_k^n)(\delta_x^+ u_k^{n+1})^2 + (\mu_x^+ u_k^{n+1})(\delta_x^+ u_k^n)^2}{2}\Bigg)\Delta x,\end{aligned}$$

for some $a \in \mathbb{R}$, typically between -1 and 2.

Remark 6.2. *We performed numerical experiments for finding a good choice of the parameter a in PDGM and based on these set $a = \frac{1}{2}$ in the following.*

Numerical tests for the Camassa–Holm equation

Example 1 (Single peakon solution): In this numerical test, we consider the same experiment as in [4], where multisymplectic schemes are considered for

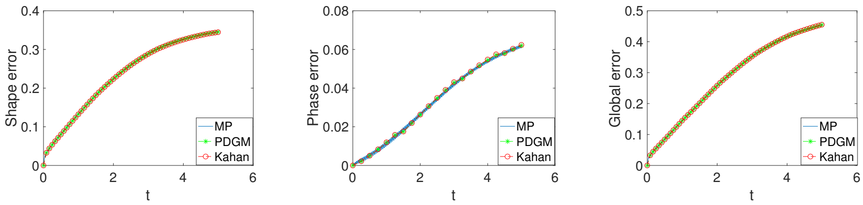


Figure 6.1: In this experiment, space step size $\Delta x = 0.04$ and time step size $\Delta t = 0.0002$. **Left:** shape error. **Middle:** phase error. **Right:** global error.

the Camassa–Holm equation with

$$u(x,0) = \frac{\cosh(|x-\frac{L}{2}|-\frac{L}{2})}{\cosh(L/2)},$$

$x \in [0,L]$, $L = 40$, $t \in [0,T]$, $T = 5$, spatial step size $\Delta x = 0.04$ and time step size $\Delta t = 0.0002$. All our methods keep a shape close to the exact solution except some small oscillatory tails, also observed in [4], resulting from the semi-discretization, see Figure 6.2 (the right two plots). The numerical simulations show that the global error is mainly due to the shape error[1], see Figure 6.1. In Figure 6.2 (the left plot), we can see that the numerical energy for all the methods oscillate, but it appears to be bounded. Here we consider also coarser grids. We observe that there appear some small wiggles for both PDGM and Kahan's method for $\Delta t = 0.02$ and long time integration $T = 100$. However, the wiggles in the solution by PDGM are much more evident than those in the solution of Kahan's method, see Figure 6.3 (the left two plots). We keep on increasing Δt to 0.15 and 0.2; we observe that the numerical solution obtained with the PDG method with $\Delta t = 0.15$ suffers from evident numerical dispersion, while Kahan's method seems to keep the shape well when comparing to the exact wave. Spurious oscillations appear also in Kahan's method when the time-step is increased to the value $\Delta t = 0.2$, see Figure 6.3 (right).

Example 2 (Two peakons solution): Now we consider the initial condition

$$u(x,0) = \frac{\cosh(|x-\frac{L}{4}|-\frac{L}{2})}{\cosh(L/2)} + \frac{3}{2}\frac{\cosh(|x-\frac{3L}{4}|-\frac{L}{2})}{\cosh(L/2)},$$

where $x \in [0,L]$, $L = 40$, $t \in [0,T]$, $T = 5$, and $\Delta x = 0.04$, $\Delta t = 0.0002$. We observe that all the methods keep the shape of the exact solution very well and the numerical energy appears bounded, see Figure 6.5. The numerical

[1] Shape error is defined by $\epsilon_{\text{shape}} := \min_{\tau} \parallel U^n - u(\cdot - \tau) \parallel_2^2$, and phase error is defined by $\epsilon_{\text{phase}} := |\underset{\tau}{\operatorname{argmin}} \parallel U^n - u(\cdot - \tau) \parallel_2^2 - ct_n|$, [5].

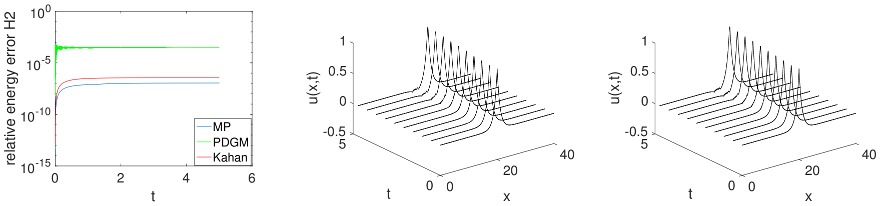


Figure 6.2: In this experiment, $\Delta x = 0.04$, $\Delta t = 0.0002$. **Left:** relative energy errors. **Middle:** propagation of the wave by PDGM. **Right:** propagation of the wave by Kahan's method.

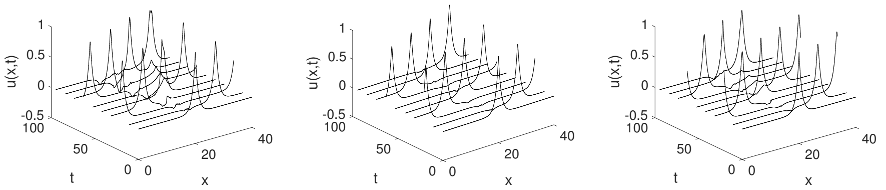


Figure 6.3: In this experiment, space step size $\Delta x = 0.04$. **Left:** propagation of the wave by PDGM, $\Delta t = 0.02$. **Middle:** propagation of the wave by Kahan's method, $\Delta t = 0.02$. **Right:** propagation of the wave by Kahan's method, $\Delta t = 0.15$.

simulation shows that the global error is mainly due to the shape error, see Figure 6.4. When a coarser time grid and longer time integration is considered, $\Delta t = 0.02$ and $T = 100$, small wiggles appear in the solution of PDGM and Kahan's method, see Figure 6.6 (the left two figures). We increase Δt to 0.2, and observe that PDGM fails to preserve the shape of the solution, while Kahan's method can still keep a shape close to the exact solution even though also for this method the numerical dispersion increases, see Figure 6.6 (right).

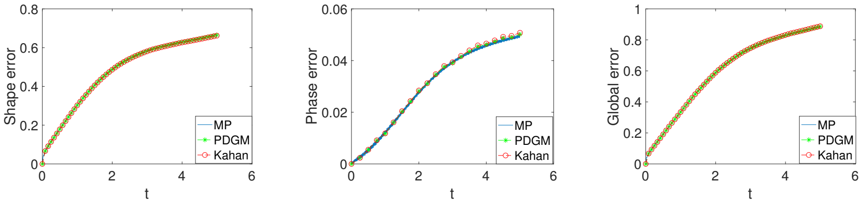


Figure 6.4: In this experiment, space step size $\Delta x = 0.04$, time step size $\Delta t = 0.0002$. **Left:** shape error. **Middle:** phase error. **Right:** global error.

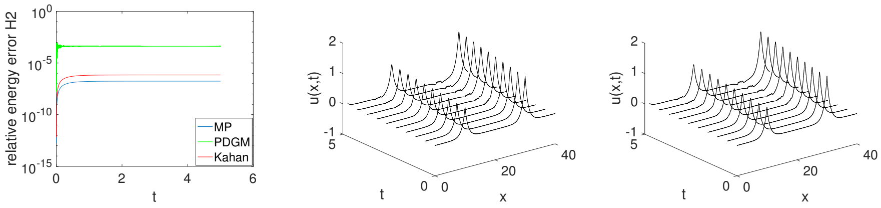


Figure 6.5: In this experiment, $\Delta x = 0.04$, $\Delta t = 0.0002$. **Left:** relative energy errors. **Middle:** propagation of the wave by PDGM. **Right:** propagation of the wave by Kahan's method.

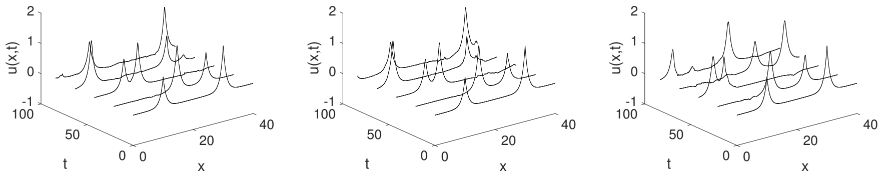


Figure 6.6: In this experiment, $\Delta x = 0.04$. **Left:** propagation of the wave by PDGM, $\Delta t = 0.02$. **Middle:** propagation of the wave by Kahan's method, $\Delta t = 0.02$. **Right:** propagation of the wave by Kahan's method, $\Delta t = 0.2$.

6.3.2 Korteweg–de Vries equation

For the Camassa–Holm equation, the vector field of the semi-discretized system is a homogeneous quadratic polynomial. In this section, we deal with the KdV equation, for which the vector field of the semi-discretized equation is a non-homogeneous quadratic polynomial. Kahan's method has also previously been used for the temporal discretization of this equation, see [13].

The KdV equation

$$u_t + 6uu_x + u_{xxx} = 0 \tag{6.15}$$

on the periodic domain $\mathbb{S} := \mathbb{R}/L\mathbb{Z}$ has the conserved Hamiltonians

$$\mathcal{H}_1(u(t)) = \frac{1}{2}\int_{\mathbb{S}} u^2 \,\mathrm{d}x, \qquad \mathcal{H}_2(u(t)) = \int_{\mathbb{S}} \left(-u^3 + \frac{1}{2}u_x^2\right) \mathrm{d}x.$$

In the following we consider the variational form based on the Hamiltonian $\mathcal{H}_2$:

$$u_t = \partial_x \frac{\delta \mathcal{H}_2}{\delta u}, \qquad \frac{\delta \mathcal{H}_2}{\delta u} = -3u^2 - u_{xx}. \tag{6.16}$$

Numerical schemes for the KdV equation

We discretize the energy $\mathcal{H}_2$ for the KdV equation (6.16) as

$$H_2(U)\Delta x = \sum_{k=1}^{K} \left(-u_k^3 + \frac{(\delta_x^+ u_k)^2 + (\delta_x^- u_k)^2}{4} \right) \Delta x.$$

From simple calculations, the corresponding gradient is given by

$$\nabla H_2(U) = \left(-3U_{\cdot}^2 - D^{\langle 2\rangle} U \right),$$

and thus we have the semi-discretized form for (6.16):

$$\dot{U} = D^{\langle 1\rangle} \left(-3U_{\cdot}^2 - D^{\langle 2\rangle} U \right). \tag{6.17}$$

Applying the schemes under consideration to (6.17) gives

$$\frac{U^{n+1} - U^n}{\Delta t} = D^{\langle 1\rangle} \nabla H_2 (\frac{U^n + U^{n+1}}{2}), \qquad \text{(MP)} \tag{6.18}$$

$$\begin{aligned} \frac{U^{n+1} - U^n}{\Delta t} =& -\frac{1}{2} D^{\langle 1\rangle} (\nabla H(U^n) + \nabla H(U^{n+1})) \\ & + 2D^{\langle 1\rangle} \nabla H(\frac{U^n + U^{n+1}}{2}), \end{aligned} \qquad \text{(Kahan)} \tag{6.19}$$

$$\frac{U^{n+2} - U^n}{2\Delta t} = D^{\langle 1\rangle} \overline{\nabla} \tilde{H}_2 (U^n, U^{n+1}, U^{n+2}), \qquad \text{(PDGM)} \tag{6.20}$$

where $H_2^{''}(U) = -6\,\mathrm{diag}(U) - D^{\langle 2\rangle}$ is the Hessian of $H_2(U)$ and the polarised discrete gradient $\overline{\nabla}\tilde{H}_2(U^n, U^{n+1}, U^{n+2})$ is found as in Proposition 6.2, with polarised discrete energy

$$\begin{aligned} \tilde{H}_2(u_k^n, u_k^{n+1})\Delta x := \sum_{k=1}^{K} (& -u_k^n u_k^{n+1} \frac{u_k^n + u_k^{n+1}}{2} + \frac{a}{2} (\delta_x^+ u_k^n)(\delta_x^+ u_k^{n+1}) \\ & + \frac{1-a}{2} \frac{(\delta_x^+ u_k^n)^2 + (\delta_x^+ u_k^{n+1})^2}{2}) \Delta x. \end{aligned}$$

Remark 6.3. *We perform several numerical simulations to find a good choice of the parameter a, and we take $a = -\frac{1}{2}$ for PDGM in the following numerical examples for the KdV equation.*

Stability analysis of the schemes

To analyse the stability of the above methods, we perform the von Neumann stability analysis for the Kahan and PDGM schemes applied to the linearized form of the KdV equation (6.15)

$$u_t + u_{xxx} = 0. \tag{6.21}$$

The equation for the amplification factor for Kahan's method is

$$(1+i\lambda(\cos\theta-1)\sin\theta)g+i\lambda(\cos\theta-1)\sin\theta-1=0,$$

and its root is

$$g=\frac{1-i\lambda(\cos\theta-1)\sin\theta}{1+i\lambda(\cos\theta-1)\sin\theta},$$

where $\lambda := \frac{\Delta t}{\Delta x^3}$. Since g is a simple root on the unit circle, Kahan's method is unconditionally stable for the linearized KdV equation.

The equation for the amplification factor for PDGM is

$$g^2-1+i\lambda(3g^2-2g+3)(\cos\theta-1)\sin\theta=0. \tag{6.22}$$

The two roots of the above equation are thus

$$g_1=\frac{3b^2+\sqrt{1+8b^2}+ib(3\sqrt{1+8b^2}-1)}{1+9b^2},$$
$$g_2=\frac{3b^2-\sqrt{1+8b^2}-ib(3\sqrt{1+8b^2}+1)}{1+9b^2},$$

where $b=\lambda(1-\cos\theta)\sin\theta$. We observe that $|g_1|=|g_2|=1$, and $g_1\neq g_2$, therefore PDGM is unconditionally stable for the linearized KdV equation.

Numerical tests for the KdV equation

Example 1 (One soliton solution): Consider the initial value

$$u(x,0)=2\,\mathrm{sech}^2(x-L/2),$$

where $x\in[0,L]$, $L=40$. We apply our schemes over the time interval $[0,T]$, $T=100$, with step sizes $\Delta x=0.05$, $\Delta t=0.0125$. From our observations, all the methods behave well. The shape of the wave is well kept by all the methods, also for long time integration, see Figure 6.7. The energy errors of all the methods are rather small and do not increase over long time integration, see Figure 6.8 (left). We then use a coarser time grid, $\Delta t=0.035$, and both methods are still stable, see Figure 6.9 (left two). However we observe that the global error of PDGM becomes much bigger than that of Kahan's method. When an even larger time step-size, $\Delta t=0.04$, is considered, the solution for PDGM blows up while the solution for Kahan's method is rather stable. In this case, the PDG method applied to the nonlinear KdV equation is unstable and the numerical solution blows up at around $t=8$. Even if we increase the time step-size to $\Delta t=0.1$, Kahan's method still works well, see Figure 6.9 (middle). When $\Delta t=0.15$ is considered, we observe evident signs of instability in the

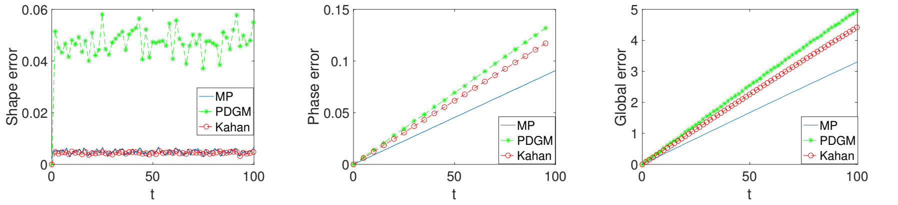


Figure 6.7: Space step size $\Delta x = 0.05$, time step size $\Delta t = 0.0125$. **Left:** shape error. **Middle:** phase error. **Right:** global error.

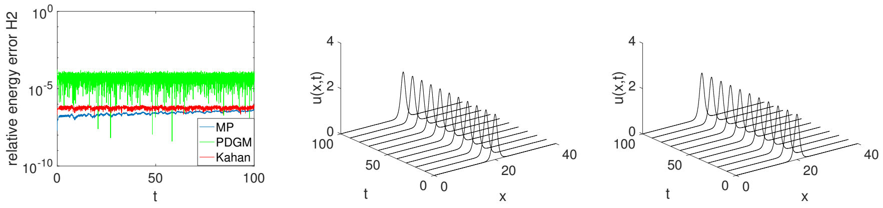


Figure 6.8: With $\Delta x = 0.05$, $\Delta t = 0.0125$. **Left:** relative energy errors. **Right two:** propagation of the wave by PDGM and Kahan's method.

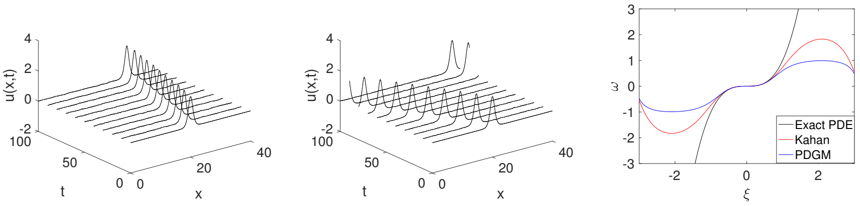


Figure 6.9: With $\Delta x = 0.05$. **Left:** $\Delta t = 0.035$, propagation of the wave by PDGM. **Middle:** $\Delta t = 0.1$, propagation of the wave by Kahan's method. **Right:** dispersion relation for $\lambda = 1$.

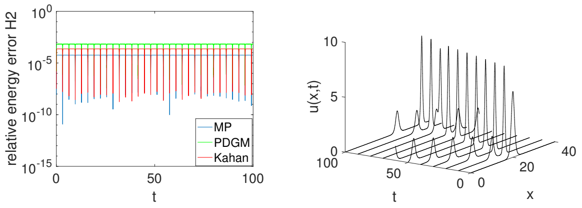


Figure 6.10: In this experiment, $\Delta x = 0.05$, $\Delta t = 0.001$. **Left:** relative energy errors. **Right two:** propagation of the wave by PDGM and Kahan's method.

solution of Kahan's method. The solution will blow up rapidly when $\Delta t = 0.2 \gg \Delta x$.

Example 2 (Two solitons solution): We choose initial value

$$u(x,0) = 6\,\mathrm{sech}^2 x,$$

and consider periodic boundary conditions $u(0,t) = u(L,t)$, where $x \in [0,L]$, $L = 40$. We set the space step size $\Delta x = 0.05$ and apply the aforementioned schemes on time interval $[0,T]$ with $T = 100$, $\Delta t = 0.001$. All the methods behave stably. The profiles of Kahan's method and the midpoint method are almost indistinguishable, and the profiles for the midpoint method are thus not presented here. Kahan's method and PDGM preserve the modified energy, and accordingly the energy error of all the methods are rather small over long time integration, see Figure 6.10 (left). After a short while the solution has two solitons; one is tall and the other is shorter, see Figure 6.10 (the right two plots).

When we consider a coarser time grid, $\Delta t = 0.00375$, both methods are still stable, see Figure 6.11 (the left two plots). However, there appear more small wiggles in the solution by PDGM and we observe that the solution of PDGM will blow up rather soon, around $t = 1$, for an even coarser time grid $\Delta t = 0.005$. When we increase the time step size to $\Delta t = 0.0125$ and consider $T = 100$, the shape of the exact solution is still well preserved by Kahan's method, even though there appear some small wiggles in the solution at around $t = 100$. We observe that the solution of Kahan's method will blow up when $\Delta t = 0.05$ is considered. Similar experiments as in this subsection, but for the multisymplectic box schemes, can be found in a paper by Ascher and McLachlan [1]. However, here we consider even coarser time grid than there, and the numerical results show that Kahan's method is quite stable, even though it is linearly implicit.

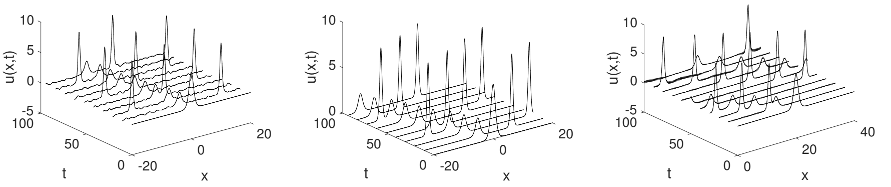


Figure 6.11: $\Delta x = 0.05$. **Left:** Propagations of the wave by PDGM, $\Delta t = 0.00375$. **Middle:** propagations of the wave by Kahan's method, $\Delta t = 0.00375$. **Right:** Propagations of the wave by Kahan's method, $\Delta t = 0.0125$.

Dispersion analysis

We consider the traditional linear analysis of numerical dispersion relations for the numerical schemes applied to the KdV equation, getting the dispersion relation of frequency ω and wave number ξ to be

$$\omega = \xi^3, \qquad \text{(exact solution)} \tag{6.23}$$

$$\sin\omega = \lambda(1-\cos\xi)(3\cos\omega - 1)\sin\xi, \qquad \text{(PDGM)} \tag{6.24}$$

$$\frac{\sin\omega}{1+\cos\omega} = \lambda(1-\cos\xi)\sin\xi, \qquad \text{(Kahan)} \tag{6.25}$$

where $\lambda = \frac{\Delta t}{\Delta x^3}$. The dispersion curve is displayed in Figure (6.9) (right). We observe that Kahan's method is better than PDGM at preserving the exact dispersion relation. This coincides with the behaviour of the methods applied to the nonlinear KdV equation shown in Section 6.3.2.

6.4 Conclusion

In this paper we perform a comparative study of Kahan's method and what we call the polarised discrete gradient (PDG) method. To that end, we present a general form encompassing a class of two-step methods that includes both a specific case of the PDG method and Kahan's method over two steps. We also compare the methods for two Hamiltonian PDEs: the KdV equation and the Camassa–Holm equation. Both Kahan's method and the PDG method are linearly implicit methods, which will save computational cost. A series of numerical experiments has been performed here, for the KdV equation with one and two solitons, and the Camassa–Holm equation with one and two peakons. These experiments show that Kahan's method is more stable than the PDG method. They also indicate that Kahan's method yields more accurate results, as we have witnessed in the energy error and the shape and phase error when comparing to analytical solutions. Based on our results, we would recommend the use of Kahan's method if one seeks a linearly implicit scheme for a Hamiltonian system with H cubic.

Acknowledgements

The authors would like to thank Elena Celledoni, Takayasu Matsuo and Brynjulf Owren for initiating the work that led to this paper, and for their very helpful comments on the manuscript.

Bibliography

[1] U. M. Ascher and R. I. McLachlan. On symplectic and multisymplectic schemes for the KdV equation. *J. Sci. Comput.*, 25(1-2):83–104, 2005.

[2] 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(20):6770–6789, 2012.

[3] E. Celledoni, R. I. McLachlan, B. Owren, and G. R. W. Quispel. Geometric properties of Kahan's method. *J. Phys. A*, 46(2):025201, 12, 2013.

[4] D. Cohen, B. Owren, and X. Raynaud. Multi-symplectic integration of the Camassa-Holm equation. *J. Comput. Phys.*, 227(11):5492–5512, 2008.

[5] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33(5):2318–2340, 2011.

[6] Z. Fei, V. M. Pérez-García, and L. Vázquez. Numerical simulation of nonlinear Schrödinger systems: a new conservative scheme. *Appl. Math. Comput.*, 71(2-3):165–177, 1995.

[7] D. Furihata. Finite difference schemes for $\partial u/\partial t = (\partial/\partial x)^\alpha \delta G/\delta u$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156(1):181–205, 1999.

[8] D. Furihata and T. Matsuo. *Discrete variational derivative method.* Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011. A structure-preserving numerical method for partial differential equations.

[9] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6(5):449–467, 1996.

[10] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[11] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 76(1):85–102, 1988.

[12] W. Kahan. Unconventional numerical methods for trajectory calculations. *Unpublished lecture notes*, 1993.

[13] W. Kahan and R.-C. Li. Unconventional schemes for a class of ordinary differential equations—with applications to the Korteweg-de Vries equation. *J. Comput. Phys.*, 134(2):316–331, 1997.

[14] T. Matsuo. New conservative schemes with discrete variational derivatives for nonlinear wave equations. *J. Comput. Appl. Math.*, 203(1):32–56, 2007.

[15] T. Matsuo and D. Furihata. Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations. *J. Comput. Phys.*, 171(2):425–447, 2001.

[16] T. Matsuo, M. Sugihara, D. Furihata, and M. Mori. Spatially accurate dissipative or conservative finite difference schemes derived by the discrete variational method. *Japan J. Indust. Appl. Math.*, 19(3):311–330, 2002.

[17] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci.*, 357(1754):1021–1045, 1999.

[18] G. R. W. Quispel and D. I. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A*, 41(4):045206, 7, 2008.

[19] M. E. Taylor. *Partial differential equations I. Basic theory*, volume 115 of *Applied Mathematical Sciences*. Springer, New York, second edition, 2011.

[20] G. Zhong and J. E. Marsden. Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators. *Phys. Lett. A*, 133(3):134–139, 1988.

Linearly implicit local and global energy-preserving methods for Hamiltonian PDEs

Sølve Eidnes and Lu Li

Submitted

Linearly implicit local and global energy-preserving methods for Hamiltonian PDEs

Abstract. We present linearly implicit methods that preserve discrete approximations to local and global energy conservation laws for multi-symplectic PDEs with cubic invariants. The methods are tested on the one-dimensional Korteweg–de Vries equation and the two-dimensional Zakharov–Kuznetsov equation; the numerical simulations confirm the conservative properties of the methods, and demonstrate their good stability properties and superior running speed when compared to fully implicit schemes.

7.1 Introduction

In recent years, much attention has been given to the design and analysis of numerical methods for differential equations that can capture geometric properties of the exact flow. The increased interest in this subject can mainly be attributed to the superior qualitative behaviour over long time integration of such structure-preserving methods, see [13, 17, 19]. A popular class of structure-preserving methods are energy-preserving methods. In particular, the energy preservation property has been found to be crucial in the proof of stability for several of these numerical methods, see e.g [16].

Energy-preserving methods are well studied for finite-dimensional Hamiltonian systems [5,7,25,30]. It is also highly conceivable that the ideas behind the finite-dimensional setting can be extended to the infinite-dimensional Hamiltonian systems or Hamiltonian partial differential equations (PDEs) [4]. There are two popular ways to construct energy-preserving methods for Hamiltonian PDEs. One approach is to semi-discretize the PDE in space so that one obtains a system of Hamiltonian ordinary differential equations (ODEs), and then apply an energy-preserving method to this semi-discrete system, see for example [7]. In this way, it is straightforward to generalise the energy-preserving methods for finite-dimensional Hamiltonian systems to Hamiltonian PDEs. However, such methods conserve only a global energy that relies on a proper boundary condition, such as a periodic boundary condition. If this is not present, the energy-preserving property will be destroyed. The other approach is based on a reformulation of the Hamiltonian PDE into a multi-symplectic form, which provides the PDE with three local conservation laws: the multi-symplectic conservation law, the energy conservation and the momentum conservation law [2, 3, 26]. Then one may consider methods that preserve the local conservation laws, see for example [34]. These locally defined properties are not dependent on the choice of boundary conditions, giving the methods that preserve local energy an advantage over methods that preserve a global energy, especially since local conservation laws will always lead to global conservation laws whenever periodic boundary conditions are considered. The concept of a multi-symplectic structure for PDEs was introduced by Bridges in [2, 3], see also [28] for a framework based on a Lagrangian formulation of the Cartan form. Local energy-preserving methods were first studied in [33], and have garnered much interest recently, see for example [18, 27, 34].

Most of the local energy-preserving methods proposed so far are fully implicit methods, for which a non-linear system must be solved at each time step. This is normally done by using an iterative solver where a linear system is solved at each iteration, which can lead to computationally expensive procedures, especially since the number of iterations needed in general increases with the size of the system. A fully explicit method on the other hand, may over-simplify the problem and often has inferior stability properties, so that a strong restriction on the grid ratio is needed. A good alternative may therefore be to develop linearly implicit schemes, where the solution at the next time step is found by solving only one linear system.

One example of linearly implicit methods for Hamiltonian ODEs is Kahan's method, which was designed for solving quadratic ODEs [24] and whose geometric properties have been studied in a series of papers by Celledoni et al. [8, 10, 11]. For Hamiltonian PDEs, Matsuo and Furihata proposed the idea of using multiple points to discretize the variational derivative and thus design linearly implicit energy-preserving schemes [29]. Dahlby and Owren generalised this concept and developed a framework for deriving linearly implicit energy-preserving multi-step methods for Hamiltonian PDEs with polynomial invariants [14]. A comparison of this approach and Kahan's method applied to PDEs is given in [15]. Recently, more work has been put into developing linearly implicit energy-preserving schemes for Hamiltonian PDEs, e.g. the partitioned averaged vector field (PAVF) method [6] and schemes based on the invariant energy quadratization (IEQ) approach [35] or the multiple scalar auxiliary variables (MSAV) approach [23]. However, little attention has been given to linearly implicit local energy-preserving methods. To the best of the authors' knowledge, the only existing method is one based on the IEQ approach, specific for the sine-Gordon equation [22]. In this paper, we use Kahan's method to construct a linearly implicit method that preserves a discrete approximation to the local energy for multi-symplectic PDEs with a cubic energy function.

The rest of this paper is organized as follows. First, we give an overview of Kahan's method and formulate it by using a polarised energy function. A brief introduction to multi-symplectic PDEs and their conservation laws are presented in Section 7.3. In Section 7.4, new linearly implicit local and global energy-preserving schemes are presented. Numerical examples for the Korteweg–de Vries (KdV) and Zakharov–Kuznetsov equations are given in Section 7.5, before we end the paper with some concluding remarks.

7.2 Kahan's method

Consider an ODE system

$$\dot{y} = f(y) = \hat{Q}(y) + \hat{B}y + \hat{c}, \quad y \in \mathbb{R}^M, \tag{7.1}$$

where $\hat{Q}(y)$ is an $\mathbb{R}^M$ valued quadratic form, $\hat{B} \in \mathbb{R}^{M\times M}$ is a symmetric constant matrix, and $\hat{c} \in \mathbb{R}^M$ is a constant vector. Kahan's method is then given by

$$\frac{y^{n+1} - y^n}{\Delta t} = \bar{Q}(y^n, y^{n+1}) + \hat{B}\frac{y^n + y^{n+1}}{2} + \hat{c},$$

where

$$\bar{Q}(y^n, y^{n+1}) = \frac{1}{2}\big(\hat{Q}(y^n + y^{n+1}) - \hat{Q}(y^n) - \hat{Q}(y^{n+1})\big)$$

is the symmetric bilinear form obtained by polarisation of the quadratic form $\hat{Q}$ [10]. Polarisation, which maps a homogeneous polynomial function to a symmetric multi-linear form in more variables, was used to generalise Kahan's method to higher degree polynomial vector fields in [9].

Suppose we restrict the problem (7.1) to be a Hamiltonian system on a Poisson vector space with a constant Poisson structure:

$$\dot{y} = A\nabla H(y), \tag{7.2}$$

where A is a constant skew-symmetric matrix, and $H : \mathbb{R}^M \to \mathbb{R}$ is a cubic polynomial function. We first consider the Hamiltonian H to be homogeneous. Then, following the result in Proposition 2.1 of [9], Kahan's method can be reformulated as

$$\frac{y^{n+1} - y^n}{\Delta t} = 3A\bar{H}(y^n, y^{n+1}, \cdot), \tag{7.3}$$

where $\bar{H}(\cdot,\cdot,\cdot) : \mathbb{R}^M \times \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}$ is a symmetric 3-tensor satisfying $\bar{H}(x,x,x) = H(x)$. Consider the 3-tensor $\bar{H}(x,y,z) = x^T Q(y) z$, where $Q(y) = \frac{1}{6}\nabla^2 H(y)$, with $\nabla^2 H$ being the Hessian of H; then we can rewrite Kahan's method (7.3) as

$$\frac{y^{n+1} - y^n}{\Delta t} = 3A\frac{\partial \bar{H}}{\partial x}\bigg|_{(y^n, y^{n+1})}, \tag{7.4}$$

where $\frac{\partial \bar{H}}{\partial x}$ denotes the partial derivative with respect to the first argument of $\bar{H}$.

Consider then the cases where the Hamiltonian in problem (7.2) is non-homogeneous, i.e. of the general form

$$H(y) = y^T Q(y) y + y^T B y + c^T y + d, \tag{7.5}$$

where $Q(y)$ is the linear part of $\nabla^2 H(y)$ and thus a symmetric matrix whose elements are homogeneous linear polynomials, B is the constant part of $\nabla^2 H(y)$ and thus a symmetric constant matrix, c is a constant vector and d is a constant scalar. We follow the technique in [10], adding one variable to $y = (y_1, \ldots, y_M)^T$ to get $\tilde{y} = (y_0, y_1, \ldots, y_M)^T$, extending A to $\tilde{A}$ by adding a zero initial row and a zero initial column, considering a homogeneous function $\tilde{H}(\tilde{y})$ based on the non-homogeneous Hamiltonian $H(y)$ such that $\tilde{H}(\tilde{y})|_{y_0=1} = H(y)$, and finally solving instead of (7.2) the equivalent, homogeneous cubic Hamiltonian problem

$$\dot{\tilde{y}} = \tilde{A} \nabla \tilde{H}(\tilde{y})$$

with $y_0 = 1$. In this way we can still get the reformulation of Kahan's method as (7.4) with

$$\begin{aligned} \bar{H}(x, y, z) = & x^T Q(y) z + \frac{1}{3}(x^T B y + y^T B z + z^T B x) \\ & + \frac{1}{3} c^T (x + y + z) + d. \end{aligned} \tag{7.6}$$

Remark 7.1. *The $\mathbb{R}$-valued function $\bar{H}(x, y, z)$ has the following properties:*

1. *$\bar{H}(x, y, z)$ is symmetric*[1] *w.r.t. x, y and z,*

2. *$\bar{H}(x, x, x) = H(x)$,*

3. *$\frac{\partial \bar{H}(x,y,z)}{\partial x} = Q(y) z + \frac{B(y+z)}{3} + \frac{c}{3}$ is symmetric w.r.t. y and z.*

In this paper, we will use the form of Kahan's method in (7.4) to prove the energy preservation of the proposed methods.

[1] Denote the elements in $Q(y)$ by $q_{ij}y = \sum_k q_{ij}^k y_k$, where q_{ij}^k, $i, j, k = 1, \cdots, M$, are scalars and y_k is the kth element of y. We have that q_{ij}^k satisfies $q_{ij}^k = q_{ki}^j = q_{jk}^i$ since $q_{ij}^k = \frac{1}{6} \frac{\partial^3 \bar{H}}{\partial y_i \partial y_j \partial y_k}$, which is unchanged under any permutation of i, j, k. This provides the symmetry of $\bar{H}(x, y, z)$.

7.3 Conservation laws for multi-symplectic PDEs

Many PDEs, including all one-dimensional Hamiltonian PDEs, can be written on the multi-symplectic form

$$Kz_t + Lz_x = \nabla S(z), \quad z \in \mathbb{R}^l, \quad (x,t) \in \mathbb{R} \times \mathbb{R}, \tag{7.7}$$

where K, $L \in \mathbb{R}^{l\times l}$ are two constant skew-symmetric matrices and $S : \mathbb{R}^l \mapsto \mathbb{R}$ is a scalar-valued function. Following the results about multi-symplectic structure in [3], it can be shown that multi-symplectic PDEs satisfy the following local conservation laws [31]: the multi-symplectic conservation law

$$\partial_t \omega + \partial_x \kappa = 0, \quad \omega = dz \wedge K_+ dz, \quad \kappa = dz \wedge L_+ dz,$$

the local energy conservation law (LECL)

$$E_t + F_x = 0, \quad E = S(z) + z_x^T L_+ z, \quad F = -z_t^T L_+ z, \tag{7.8}$$

and the local momentum conservation law (LMCL)

$$I_t + G_x = 0, \quad G = S(z) + z_t^T K_+ z, \quad I = -z_x^T K_+ z,$$

where K_+ and L_+ satisfy

$$K = K_+ - K_+^T, \quad L = L_+ - L_+^T.$$

Decomposition of the matrices is done to make deduction of the conservations laws for energy and momentum more efficient [26, Section 12.3.1].

The multi-symplectic form (7.7) can also be generalised to problems in higher dimensional spaces. Consider d spatial dimensions; based on the work by Bridges [3], a multi-symplectic PDE can then be written as

$$Kz_t + \sum_{\alpha=1}^{d} L^\alpha z_{x_\alpha} = \nabla S(z), \quad z \in \mathbb{R}^l, \quad (x,t) \in \mathbb{R}^d \times \mathbb{R}, \tag{7.9}$$

where K, $L^\alpha \in \mathbb{R}^{l\times l}$ ($\alpha = 1, \ldots, d$) are constant skew-symmetric matrices and $S : \mathbb{R}^l \to \mathbb{R}$ is a smooth functional. Equation (7.9) has the following local energy conservation law:

$$E_t + \sum_{\alpha=1}^{d} F^\alpha_{x_\alpha} = 0, \tag{7.10}$$

where $E(z) = S(z) + \sum_{\alpha=1}^{d} z_\alpha^T L_+^\alpha z$, $F^\alpha = -z_t^T L_+^\alpha z$, and L_+^α are splittings of L^α satisfying $L^\alpha = L_+^\alpha - (L_+^\alpha)^T$.

Say we have (7.9) defined on the spatial domain $\Omega \in \mathbb{R}^d$ with periodic boundary conditions. Integrating over the domain Ω on both sides of the equation (7.10) and using the periodic boundary condition then leads to the global energy conservation law for the multi-symplectic PDEs,

$$\frac{d}{dt}\mathcal{E}(z) = 0, \tag{7.11}$$

where $\mathcal{E}(z) = \int_\Omega E(z)d\Omega$.

Example 7.1. ***Korteweg–de Vries equation.*** *Consider the KdV equation for modeling shallow water waves,*

$$u_t + \eta u u_x + \gamma^2 u_{xxx} = 0, \tag{7.12}$$

where $\eta, \gamma \in \mathbb{R}$. Introducing the potential $\phi_x = u$, momenta $v = \gamma u_x$ and the variable $w = \gamma v_x \phi_t + \frac{\gamma^2 u^2}{2}$ by the covariant Legendre transform from the Lagrangian, we obtain

$$\begin{aligned} \frac{1}{2}u_t + w_x &= 0, \\ -\frac{1}{2}\phi_t - \gamma v_x &= -w + \frac{\eta}{2}u^2, \\ \gamma u_x &= v, \\ -\phi_x &= -u, \end{aligned} \tag{7.13}$$

from which we find the multi-symplectic formulation (7.7) *for the KdV equation with $z = (\phi, u, v, w)^T$, the Hamiltonian $S(z) = \frac{v^2}{2} - uw + \frac{\eta u^3}{6}$, and*

$$K = \begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 \\ -\frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad L = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -\gamma & 0 \\ 0 & \gamma & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}.$$

As for the conservation laws, there are many choices of K_+ and L_+, for example $K_+ = \frac{K}{2}, L_+ = \frac{L}{2}$, or K_+ and L_+ being the upper triangular parts of K and L, respectively.

Example 7.2. ***Zakharov–Kuznetsov equation.*** *Zakharov and Kuznetsov introduced in [37] a (2+1)-dimensional generalisation of the KdV equation which includes weak transverse variation,*

$$u_t + uu_x + u_{xxx} + u_{xyy} = 0. \tag{7.14}$$

A multi-symplectification of this leads to a system (7.9) for two spatial dimensions,

$$Kz_t + L^1 z_x + L^2 z_y = \nabla S(z), \quad z \in \mathbb{R}^6, \quad (x, y, t) \in \mathbb{R}^2 \times \mathbb{R}. \tag{7.15}$$

Following [4], we have that (7.14) *is equivalent to a system of first-order PDEs,*

$$\begin{aligned} \phi_x &= u, \\ \frac{1}{2}\phi_t + v_x + w_y &= p - \frac{1}{2}u^2, \\ w_x - v_y &= 0, \\ -\frac{1}{2}u_t - p_x &= 0, \\ -u_x + q_y &= -v, \\ -q_x - u_y &= -w, \end{aligned} \tag{7.16}$$

which is (7.15) *with* $z = (p, u, q, \phi, v, w)^T$*, the Hamiltonian* $S(z) = up - \frac{1}{2}(v^2 + w^2) - \frac{1}{6}u^3$*, and the skew-symmetric matrices*

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

$$L^1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}, \quad L^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

7.4 New linearly implicit energy-preserving schemes

In [18], Gong, Cai and Wang present a scheme that preserves the local energy conservation law (7.8) of a one-dimensional multi-symplectic PDE, obtained by applying the midpoint rule in space and the averaged vetor field (AVF) method in time. They also present schemes that preserve the global energy, but not (7.8), obtained by considering spatial discretizations that preserve the skew-symmetric property of the difference operator ∂_x. We build on their work by considering Kahan's method for the discretization in time, ensuring linearly implicit schemes and also energy preservation.

To introduce our new schemes, we begin with some basic difference operators:

$$\delta_t v_j^n := \frac{v_j^{n+1} - v_j^n}{\Delta t}, \qquad \delta_x v_j^n := \frac{v_{j+1}^n - v_j^n}{\Delta x}$$
$$\mu_t v_j^n := \frac{v_j^{n+1} + v_j^n}{2}, \qquad \mu_x v_j^n := \frac{v_{j+1}^n + v_j^n}{2}.$$

The operators satisfy the following properties [34]:

1. All the operators commute with each other, e.g.
$$\delta_t\delta_x v_j^n = \delta_x\delta_t v_j^n, \quad \delta_t\mu_x v_j^n = \mu_x\delta_t v_j^n, \quad \mu_t\delta_x v_j^n = \delta_x\mu_t v_j^n.$$
2. They satisfy the discrete Leibniz rule
$$\delta_t(uv)_j^n = (\varepsilon u_j^{n+1}+(1-\varepsilon)u_j^n)\delta_t v_j^n+\delta_t u_j^n((1-\varepsilon)v_j^{n+1}+\varepsilon v_j^n), \quad 0\leq\varepsilon\leq 1.$$
Specifically,
$$\begin{aligned}\delta_t(uv)_j^n &= u_j^n\delta_t v_j^n+\delta_t u_j^n v_j^{n+1}, &&\text{for} \quad \varepsilon=0,\\ \delta_t(uv)_j^n &= \mu_t u_j^n\delta_t v_j^n+\delta_t u_j^n\mu_t v_j^n, &&\text{for} \quad \varepsilon=\frac{1}{2},\\ \delta_t(uv)_j^n &= u_j^{n+1}\delta_t v_j^n+\delta_t u_j^n v_j^n, &&\text{for} \quad \varepsilon=1.\end{aligned}$$

One can obtain a series of similar commutative equations and discrete Leibniz rules that are not presented here, but which are also crucial in the proofs of the preservation properties of the schemes to be introduced in the remainder of this section.

7.4.1 A local energy-preserving scheme for multi-symplectic PDEs

In this section, we apply the midpoint rule in space and Kahan's method in time to construct a local energy-preserving method for multi-symplectic PDEs. Introducing the concept by first considering the one-dimensional system (7.7), we apply the midpoint rule in space to get

$$K\partial_t\mu_x z_j + L\delta_x z_j = \nabla S(\mu_x z_j), \quad j=0,\ldots,M-1.$$

Then applying Kahan's method gives us the linearly implicit local energy-preserving (LILEP) scheme

$$K\delta_t\mu_x z_j^n + L\delta_x\mu_t z_j^n = 3\frac{\partial\bar{S}}{\partial x}\bigg|_{(\mu_x z_j^n,\mu_x z_j^{n+1})}. \tag{7.17}$$

Here we consider S of the form $S(y) = y^T Q(y)y + y^T By + c^T y + d$, as in (7.5), and accordingly $\bar{S}(x,y,z)$ of the form (7.6).

Theorem 7.1. *The scheme* (7.17) *satisfies the discrete local energy conservation law*

$$\delta_t(\bar{E}_L)_j^n + \delta_x(\bar{F}_L)_j^n = 0, \tag{7.18}$$

where

$$\begin{aligned}(\bar{E}_L)_j^n &= \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1}) + \frac{1}{3}(\delta_x z_j^n)^T L_+ \mu_x z_j^n \\ &\quad + \frac{1}{3}(\delta_x z_j^n)^T L_+ \mu_x z_j^{n+1} + \frac{1}{3}(\delta_x z_j^{n+1})^T L_+ \mu_x z_j^n,\end{aligned} \tag{7.19}$$

$$(\bar{F}_L)_j^n = -\frac{1}{3}(\delta_t z_j^n)^T L_+ \mu_t z_j^n - \frac{1}{3}(\delta_t z_j^n)^T L_+ \mu_t z_j^{n+1} - \frac{1}{3}(\delta_t z_j^{n+1})^T L_+ \mu_t z_j^n.$$

Proof. Taking the inner product with $\frac{1}{3}\delta_t \mu_x z_j^n$ on both sides of (7.17) and using the skew-symmetry of matrix K, we have

$$\frac{1}{3}(\delta_t \mu_x z_j^n)^T L \delta_x \mu_t z_j^n = (\delta_t \mu_x z_j^n)^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(\mu_x z_j^n, \mu_x z_j^{n+1})}.$$

Taking the inner product with $\frac{1}{3}\delta_t \mu_x z_j^{n+1}$ on both sides of (7.17), we get

$$\frac{1}{3}(\delta_t \mu_x z_j^{n+1})^T K \delta_t \mu_x z_j^n + \frac{1}{3}(\delta_t \mu_x z_j^{n+1})^T L \delta_x \mu_t z_j^n = (\delta_t \mu_x z_j^{n+1})^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(\mu_x z_j^n, \mu_x z_j^{n+1})}.$$

Taking the inner product with $\frac{1}{3}\delta_t \mu_x z_j^n$ on both sides of the scheme (7.17) for the next time step, we get

$$\frac{1}{3}(\delta_t \mu_x z_j^n)^T K \delta_t \mu_x z_j^{n+1} + \frac{1}{3}(\delta_t \mu_x z_j^n)^T L \delta_x \mu_t z_j^{n+1} = (\delta_t \mu_x z_j^n)^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(\mu_x z_j^{n+1}, \mu_x z_j^{n+2})}.$$

Adding the last three equations and using the skew-symmetry of matrix K, we obtain

$$\begin{aligned}&\frac{1}{3}\Big((\delta_t \mu_x z_j^n)^T L \delta_x \mu_t z_j^n + (\delta_t \mu_x z_j^{n+1})^T L \delta_x \mu_t z_j^n + (\delta_t \mu_x z_j^n)^T L \delta_x \mu_t z_j^{n+1}\Big) \\ &= (\delta_t \mu_x z_j^n)^T \frac{\partial \bar{S}}{\partial x}|_{(\mu_x z_j^n, \mu_x z_j^{n+1})} + (\delta_t \mu_x z_j^{n+1})^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(\mu_x z_j^n, \mu_x z_j^{n+1})} \\ &\quad + (\delta_t \mu_x z_j^n)^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(\mu_x z_j^{n+1}, \mu_x z_j^{n+2})}, \\ &= \frac{1}{\Delta t}\big(\bar{S}(\mu_x z_j^{n+1}, \mu_x z_j^{n+1}, \mu_x z_j^{n+2}) - \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1})\big), \\ &= \delta_t \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1}).\end{aligned} \tag{7.20}$$

On the other hand, using the aforementioned commutative laws and discrete Leibniz rules for the operators, we can deduce

$$\begin{aligned}
\delta_t((\delta_x z_j^n)^T L_+ \mu_x z_j^n) &= (\delta_t\delta_x z_j^n)^T L_+ \mu_t\mu_x z_j^n + (\delta_x\mu_t z_j^n)^T L_+ \delta_t\mu_x z_j^n,\\
\delta_x((\delta_t z_j^n)^T L_+ \mu_t z_j^n) &= (\delta_t\delta_x z_j^n)^T L_+ \mu_t\mu_x z_j^n + (\delta_t\mu_x z_j^n)^T L_+ \delta_x\mu_t z_j^n,\\
\delta_t((\delta_x z_j^{n+1})^T L_+ \mu_x z_j^n) &= (\delta_t\delta_x z_j^{n+1})^T L_+ \mu_t\mu_x z_j^n + (\delta_x\mu_t z_j^{n+1})^T L_+ \delta_t\mu_x z_j^n,\\
\delta_x((\delta_t z_j^{n+1})^T L_+ \mu_t z_j^n) &= (\delta_t\delta_x z_j^{n+1})^T L_+ \mu_t\mu_x z_j^n + (\delta_t\mu_x z_j^{n+1})^T L_+ \delta_x\mu_t z_j^n,\\
\delta_t((\delta_x z_j^n)^T L_+ \mu_x z_j^{n+1}) &= (\delta_t\delta_x z_j^n)^T L_+ \mu_t\mu_x z_j^{n+1} + (\delta_x\mu_t z_j^n)^T L_+ \delta_t\mu_x z_j^{n+1},\\
\delta_x((\delta_t z_j^n)^T L_+ \mu_t z_j^{n+1}) &= (\delta_t\delta_x z_j^n)^T L_+ \mu_t\mu_x z_j^{n+1} + (\delta_t\mu_x z_j^n)^T L_+ \delta_x\mu_t z_j^{n+1}.
\end{aligned}$$

Using the above relations, the fact that $L = L_+ - L_+^T$ and the result (7.20), we obtain

$$\begin{aligned}
\delta_t E_j^n + \delta_x F_j^n =& \, \delta_t \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1})\\
&+ \frac{1}{3}\big(\delta_t((\delta_x z_j^n)^T L_+ \mu_x z_j^n) + \delta_t((\delta_x z_j^n)^T L_+ \mu_x z_j^{n+1})\\
&+ \delta_t((\delta_x z_j^{n+1})^T L_+ \mu_x z_j^n)\big) - \frac{1}{3}\big(\delta_x((\delta_t z_j^n)^T L_+ \mu_t z_j^n)\\
&+ \delta_x((\delta_t z_j^n)^T L_+ \mu_t z_j^{n+1}) + \delta_x((\delta_t z_j^{n+1})^T L_+ \mu_t z_j^n)\big)\\
=& \, \delta_t \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1}) - \frac{1}{3}\big((\delta_t\mu_x z_j^n)^T L \delta_x\mu_t z_j^n\\
&+ (\delta_t\mu_x z_j^{n+1})^T L \delta_x\mu_t z_j^n + (\delta_t\mu_x z_j^n)^T L \delta_x\mu_t z_j^{n+1}\big)\\
=& \, 0.
\end{aligned}$$

□

Corollary 2. For periodic boundary conditions $z(x+P,t) = z(x,t)$, the scheme (7.17) satisfies the discrete global energy conservation law

$$\bar{\mathcal{E}}_L^{n+1} = \bar{\mathcal{E}}_L^n, \quad \bar{\mathcal{E}}_L^n := \Delta x \sum_{j=0}^{M-1} (\bar{E}_L)_j^n, \tag{7.21}$$

where $\Delta x = P/M$ and $(\bar{E}_L)_j^n$ is given by (7.19).

Proof. With periodic boundary conditions, we get $\sum_{j=0}^{M-1} \delta_x (\bar{F}_L)_j^n = 0$, and thus (7.21) follows from (7.18). □

The polarised global energy $\bar{\mathcal{E}}_L^n$ may be considered as a function of the solution in time step n only, similarly to the modified Hamiltonian defined in Proposition 3 of [10].

Proposition 7.1. *With the solution* z^{n+1} *found from* z^n *by* (7.17), *the discrete global energy* $\bar{\mathcal{E}}_L^n$ *of* (7.21) *satisfies*

$$\bar{\mathcal{E}}_L^n = \mathcal{E}_L^n + \Delta x \sum_{j=0}^{M-1} \frac{1}{3} (\nabla E_L(z_j^n))^T (z_j^{n+1} - z_j^n), \tag{7.22}$$

where

$$\mathcal{E}_L^n := \Delta x \sum_{j=0}^{M-1} E_L(z_j^n), \quad E_L(z_j^n) := S(\mu_x z_j^n) + (\delta_x z_j^n)^T L_+ \mu_x z_j^n, \tag{7.23}$$

while $z_j^{n+1} - z_j^n$ *satisfies*

$$R_L(z_j^n)(z_j^{n+1} - z_j^n) = \Delta t g_L(z_j^n), \tag{7.24}$$

with $g_L(z_j^n) = \nabla S(\mu_x z_j^n) - L\delta_x z_j^n$ *and* $R_L(z_j^n) = K\mu_x - \frac{\Delta t}{2} \nabla g_L(z_j^n)$.

Proof. Note that

$$\begin{aligned} \bar{S}(\mu_x z_j^n, \mu_x z_j^n, \mu_x z_j^{n+1}) &= S(\mu_x z_j^n) + \frac{1}{3} \nabla S(\mu_x z_j^n)^T (\mu_x z_j^{n+1} - \mu_x z_j^n) \\ &= S(\mu_x z_j^n) + \frac{1}{3} \nabla_{z_j^n} (S(\mu_x z_j^n))^T (z_j^{n+1} - z_j^n), \end{aligned}$$

and

$$\begin{aligned} &\frac{1}{3} (\delta_x z_j^n)^T L_+ \mu_x z_j^n + \frac{1}{3} (\delta_x z_j^n)^T L_+ \mu_x z_j^{n+1} + \frac{1}{3} (\delta_x z_j^{n+1})^T L_+ \mu_x z_j^n \\ &= (\delta_x z_j^n)^T L_+ \mu_x z_j^n \\ &\quad + \frac{1}{3} \big((\delta_x z_j^n)^T L_+ (\mu_x z_j^{n+1} - \mu_x z_j^n) + (\delta_x z_j^{n+1} - \delta_x z_j^n)^T L_+ \mu_x z_j^n \big) \\ &= (\delta_x z_j^n)^T L_+ \mu_x z_j^n + \frac{1}{3} \big((\mu_x z_j^n)^T L_x^T \delta_x + (\delta_x z_j^n)^T L_x \mu_x \big) (z_j^{n+1} - z_j^n) \\ &= (\delta_x z_j^n)^T L_+ \mu_x z_j^n + \frac{1}{3} \Big(\nabla_{z_j^n} \big((\delta_x z_j^n)^T L_+ \mu_x z_j^n \big) \Big)^T (z_j^{n+1} - z_j^n). \end{aligned}$$

Inserting this in (7.19), we get (7.22) from (7.21). Furthermore, observing that

$$3 \frac{\partial \bar{S}}{\partial x} \bigg|_{(\mu_x z_j^n, \mu_x z_j^{n+1})} = \nabla S(\mu_x z_j^n) + \frac{1}{2} \nabla^2 S(\mu_x z_j^n)(\mu_x z_j^{n+1} - \mu_x z_j^n),$$

we may rewrite (7.17) as

$$\Big(K\mu_x + \frac{\Delta t}{2} L\delta_x - \frac{\Delta t}{2} \nabla^2 S(\mu_x z_j^n) \mu_x \Big) (z_j^{n+1} - z_j^n) = \Delta t \big(\nabla S(\mu_x z_j^n) - L\delta_x z_j^n \big),$$

which is (7.24). □

Note that (7.23) is the discrete energy preserved by the fully implicit local energy-preserving method of [18]. Also, for methods based on the multi-symplectic structure, instead of solving for z directly, the normal procedure is to eliminate the auxiliary variables from the scheme and get an equation for one variable u. Therefore we do not give an explicit expression for the modified energy in z^n. However, in Section 7.5, we present an explicit expression for the modified energy in u^n when our scheme is applied to the KdV equation.

The results about the energy conservation for the LILEP method applied to one-dimensional multi-symplectic PDEs can be generalised to problems in spatial dimensions of any finite degree. Consider for example a 2-dimensional multi-symplectic PDE

$$Kz_t + L^1 z_x + L^2 z_y = \nabla S(z), \quad z \in \mathbb{R}^l, \quad (x,y,t) \in \mathbb{R}^3, \tag{7.25}$$

for which we have the following corollary. This is presented without its proof, which is rather technical but similar to the proof of Theorem 7.1.

Corollary 3. The scheme obtained by applying the midpoint rule in space and Kahan's method in time to equation (7.25),

$$K\delta_t\mu_x\mu_y z^n_{j,k} + L^1\delta_x\mu_t\mu_y z^n_{j,k} + L^2\delta_y\mu_t\mu_x z^n_{j,k} = 3\left.\frac{\partial\bar{S}}{\partial x}\right|_{(\mu_x\mu_y z^n_{j,k},\mu_x\mu_y z^{n+1}_{j,k})},$$

where $j = 0,\dots,M_x - 1$ and $k = 0,\dots,M_y - 1$, satisfies the discrete local energy conservation law

$$\delta_t(\bar{E}_L)^n_{j,k} + \delta_x(\bar{F}^1_L)^n_{j,k} + \delta_y(\bar{F}^2_L)^n_{j,k} = 0,$$

where

$$\begin{aligned}
(\bar{E}_L)^n_{j,k} = {}& \bar{S}(\mu_x\mu_y z^n_{j,k}, \mu_x\mu_y z^n_{j,k}, \mu_x\mu_y z^{n+1}_{j,k}) \\
& + \frac{1}{3}(\delta_x\mu_y z^n_{j,k})^T L^1_+ \mu_x\mu_y z^n_{j,k} + \frac{1}{3}(\delta_x\mu_y z^n_{j,k})^T L^1_+ \mu_x\mu_y z^{n+1}_{j,k} \\
& + \frac{1}{3}(\delta_x\mu_y z^{n+1}_{j,k})^T L^1_+ \mu_x\mu_y z^n_{j,k} + \frac{1}{3}(\delta_y\mu_x z^n_{j,k})^T L^2_+ \mu_x\mu_y z^n_{j,k} \\
& + \frac{1}{3}(\delta_y\mu_x z^n_{j,k})^T L^2_+ \mu_x\mu_y z^{n+1}_{j,k} + \frac{1}{3}(\delta_y\mu_x z^{n+1}_{j,k})^T L^2_+ \mu_x\mu_y z^n_{j,k}, \\
(\bar{F}^1_L)^n_{j,k} = {}& -\frac{1}{3}(\delta_t\mu_y z^n_{j,k})^T L^1_+ \mu_t\mu_y z^n_{j,k} - \frac{1}{3}(\delta_t\mu_y z^n_{j,k})^T L^1_+ \mu_t\mu_y z^{n+1}_{j,k} \\
& - \frac{1}{3}(\delta_t\mu_y z^{n+1}_{j,k})^T L^1_+ \mu_t\mu_y z^n_{j,k}, \\
(\bar{F}^2_L)^n_{j,k} = {}& -\frac{1}{3}(\delta_t\mu_x z^n_{j,k})^T L^2_+ \mu_t\mu_x z^n_{j,k} - \frac{1}{3}(\delta_t\mu_x z^n_{j,k})^T L^2_+ \mu_t\mu_x z^{n+1}_{j,k} \\
& - \frac{1}{3}(\delta_t\mu_x z^{n+1}_{j,k})^T L^2_+ \mu_t\mu_x z^n_{j,k}.
\end{aligned}$$

7.4.2 Global energy-preserving methods for multi-symplectic PDEs

As shown in Section 7.3, Hamiltonian PDEs of the form (7.7) with periodic boundary conditions have global energy conservation which can be deduced from the local conservation law. On the other hand, the local conservation law is not inherent in the global conservation law. In this section, we will focus on giving a systematic method that preserves the global energy conservation law directly. We discretize ∂_x with an antisymmetric differential matrix D and get the semi-discretized variant of (7.7),

$$K\partial_t z_j + L(Dz)_j = \nabla S(z_j), \quad j = 0, 1, \dots, M-1, \tag{7.26}$$

where $z := (z_0, z_1, \dots, z_{M-1})^T \in \mathbb{R}^{M\times l}$ and $(Dz)_j = \sum_{k=0}^{M-1} D_{j,k} z_k$. We then apply Kahan's method to (7.26) and obtain the linearly implicit global energy-preserving (LIGEP) scheme

$$K\delta_t z_j^n + L(D\mu_t z^n)_j = 3\left.\frac{\partial \bar{S}}{\partial x}\right|_{(z_j^n, z_j^{n+1})}. \tag{7.27}$$

Define the polarised energy density by

$$\begin{aligned}\bar{E}_j^n = {} & \bar{S}(z_j^n, z_j^n, z_j^{n+1}) + \frac{1}{3}(Dz^n)_j^T L_+ z_j^n \\ & + \frac{1}{3}(Dz^n)_j^T L_+ z_j^{n+1} + \frac{1}{3}(Dz^{n+1})_j^T L_+ z_j^n,\end{aligned} \tag{7.28}$$

and we get the following result.

Theorem 7.2. *For periodic boundary conditions $z(x+P, t) = z(x, t)$, the scheme* (7.27) *satisfies the discrete global energy conservation law*

$$\bar{\mathcal{E}}^{n+1} = \bar{\mathcal{E}}^n, \quad \bar{\mathcal{E}}^n := \Delta x \sum_{j=0}^{M-1} \bar{E}_j^n, \quad \Delta x = P/M. \tag{7.29}$$

Proof. Taking the inner product with $\frac{1}{3}\delta_t z_j^n$ on both sides of equation (7.27) and using the skew-symmetry of the matrix K, we get

$$\frac{1}{3}(\delta_t z_j^n)^T L(D\mu_t z^n)_j = (\delta_t z_j^n)^T \left.\frac{\partial \bar{S}}{\partial x}\right|_{(z_j^n, z_j^{n+1})}. \tag{7.30}$$

Taking the inner product with $\frac{1}{3}\delta_t z_j^{n+1}$ on both sides of (7.27), we get

$$\frac{1}{3}(\delta_t z_j^{n+1})^T K\delta_t z_j^n + \frac{1}{3}(\delta_t z_j^{n+1})^T L(D\mu_t z^n)_j = (\delta_t z_j^{n+1})^T \left.\frac{\partial \bar{S}}{\partial x}\right|_{(z_j^n, z_j^{n+1})}. \tag{7.31}$$

Furthermore, taking the inner product with $\frac{1}{3}\delta_t z_j^n$ on both sides of (7.27) for the next time step, we have

$$\frac{1}{3}(\delta_t z_j^n)^T K \delta_t z_j^{n+1} + \frac{1}{3}(\delta_t z_j^n)^T L(D\mu_t z^{n+1})_j = (\delta_t z_j^n)^T \frac{\partial \bar{S}}{\partial x}\bigg|_{(z_j^{n+1}, z_j^{n+2})}. \tag{7.32}$$

Adding equations (7.30), (7.31) and (7.32), we get

$$\begin{aligned}\frac{1}{3}\big((\delta_t z_j^n)^T L(D\mu_t z^n)_j + (\delta_t z_j^n)^T L(D\mu_t z^{n+1})_j \\ + (\delta_t z_j^{n+1})^T L(D\mu_t z^n)_j\big) = \delta_t \bar{S}(z_j^n, z_j^n, z_j^{n+1}).\end{aligned} \tag{7.33}$$

By using the commutative laws and discrete Leibniz rules,

$$\begin{aligned}\delta_t((Dz^n)_j^T L_+ z_j^n) &= (D\delta_t z^n)_j^T L_+ \mu_t z_j^n + (D\mu_t z^n)_j L_+ \delta_t z_j^n,\\ \delta_t((Dz^n)_j^T L_+ z_j^{n+1}) &= (D\delta_t z^n)_j^T L_+ \mu_t z_j^{n+1} + (D\mu_t z^n)_j L_+ \delta_t z_j^{n+1},\\ \delta_t((Dz^{n+1})_j^T L_+ z_j^n) &= (D\delta_t z^{n+1})_j^T L_+ \mu_t z_j^n + (D\mu_t z^{n+1})_j L_+ \delta_t z_j^n.\end{aligned} \tag{7.34}$$

Based on the above equations (7.33) and (7.34), we obtain

$$\begin{aligned}\delta_t E_j^n =& \delta_t \bar{S}(z_j^n, z_j^n, z_j^{n+1})\\ &+ \frac{1}{3}\big(\delta_t((Dz^n)_j^T L_+ z_j^n) + (Dz^n)_j^T L_+ z_j^{n+1} + (Dz^{n+1})_j^T L_+ z_j^n\big)\\ =& \frac{1}{3}\big((\delta_t z_j^n)^T L_+ (D\mu_t z^n)_j + (D\delta_t z^n)_j^T L_+ \mu_t z_j^n\big)\\ &+ \frac{1}{3}\big((\delta_t z_j^{n+1})^T L_+ (D\mu_t z^n)_j + (D\delta_t z^{n+1})_j^T L_+ \mu_t z_j^n\big)\\ &+ \frac{1}{3}\big((\delta_t z_j^n)^T L_+ (D\mu_t z^{n+1})_j + (D\delta_t z^n)_j^T L_+ \mu_t z_j^{n+1}\big)\\ =& \sum_{k=0}^{N-1} (D)_{j,k} G_{j,k},\end{aligned}$$

where

$$\begin{aligned}G_{j,k} :=& \frac{1}{3}\big((\delta_t z^n)_j^T L_+ \mu_t z_L^n + (\delta_t z^n)_L^T L_+ \mu_t z_j^n\big)\\ &+ \frac{1}{3}\big((\delta_t z^{n+1})_j^T L_+ \mu_t z_L^n + (\delta_t z^{n+1})_L^T L_+ \mu_t z_j^n\big)\\ &+ \frac{1}{3}\big((\delta_t z^n)_j^T L_+ \mu_t z_L^{n+1} + (\delta_t z^n)_L^T L_+ \mu_t z_j^{n+1}\big).\end{aligned}$$

Since D is skew-symmetric and $G_{j,k} = G_{k,j}$, we get

$$\sum_{j=0}^{M-1} \delta_t \bar{E}_j^n = 0,$$

which implies that the discrete global energy conservation law $\bar{\mathcal{E}}^{n+1} = \bar{\mathcal{E}}^n$ is satisfied. □

The polarised energy $\bar{\mathcal{E}}$ preserved by (7.27) may also be expressed as a modification of the discrete energy

$$\mathcal{E}^n := \Delta x \sum_{j=0}^{M-1} E(z_j^n), \quad E(z_j^n) = S(z_j^n) + (Dz^n)_j^T L_+ z_j^n, \tag{7.35}$$

which is preserved by the fully implicit global energy-preserving scheme of [18]. The proof of the following proposition is similar to the proof of Proposition 7.1, and hence omitted.

Proposition 7.2. *If the solution z^{n+1} is found from z^n by* (7.27)*, the discrete global energy $\bar{\mathcal{E}}^n$ of* (7.29) *satisfies*

$$\bar{\mathcal{E}}^n = \mathcal{E}^n + \Delta x \sum_{j=0}^{M-1} \frac{1}{3} (\nabla E(z_j^n))^T (z_j^{n+1} - z_j^n),$$

and $z_j^{n+1} - z_j^n$ satisfies

$$R(z_j^n)(z_j^{n+1} - z_j^n) = \Delta t g(z_j^n),$$

where $g(z_j^n) = \nabla S(z_j^n) - L(Dz)_j^n$ and $R(z_j^n) = K + \frac{\Delta t}{2} \nabla g(z_j^n)$.

The above global conservation results can be generalised to multi-symplectic formulations in higher spatial dimensions, as demonstrated for the two-dimensional case by the following corollary, whose omitted proof is in the same vein as the proof of Theorem 7.2.

Corollary 4. Discretizing ∂_x and ∂_y by skew-symmetric differential matrices D_x and D_y in equation (7.25) and then applying Kahan's method to the semi-discrete system, one obtains the linearly implicit global energy-preserving (LIGEP) scheme

$$K\delta_t z_{j,k}^n + L^1 \mu_t (D_x z^n)_{j,k} + L^2 \mu_t (D_y z^n)_{j,k} = 3 \frac{\partial \bar{S}}{\partial x}\bigg|_{(z_{j,k}^n, z_{j,k}^{n+1})}, \tag{7.36}$$

where $j = 0, \dots, M_x - 1$ and $k = 0, \dots, M_y - 1$. For periodic boundary conditions $z(x + P_x, y, t) = z(x, y, t)$, $z(x, y + P_y, t) = z(x, y, t)$, the scheme (7.36) satisfies the discrete global energy conservation law

$$\bar{\mathcal{E}}^{n+1} = \bar{\mathcal{E}}^n,$$

where

$$\begin{aligned}
\bar{\mathcal{E}}^n &:= \Delta x \Delta y \sum_{j=0}^{M_x-1} \sum_{k=0}^{M_y-1} \bar{E}^n_{j,k}, \quad \Delta x = P_x/M_x, \quad \Delta y = P_y/M_y, \\
\bar{E}^n_{j,k} &= \bar{S}(z^n_{j,k}, z^n_{j,k}, z^{n+1}_{j,k}) + \frac{1}{3}(D_x z^n)^T_{j,k} L^1_+ z^n_{j,k} \\
&\quad + \frac{1}{3}(D_x z^n)^T_{j,k} L^1_+ z^{n+1}_{j,k} + \frac{1}{3}(D_x z^{n+1})^T_{j,k} L^1_+ z^n_{j,k}, \\
&\quad + \frac{1}{3}(D_y z^n)^T_{j,k} L^2_+ z^n_{j,k} + \frac{1}{3}(D_y z^n)^T_{j,k} L^2_+ z^{n+1}_{j,k} + \frac{1}{3}(D_y z^{n+1})^T_{j,k} L^2_+ z^n_{j,k}.
\end{aligned}$$

7.5 Numerical examples

In this section, we apply our proposed new linearly implicit energy-preserving schemes to the KdV equation and Zakharov–Kuznetsov equation, and compare them with fully implicit schemes. Among our reference methods are the methods introduced in [18], for which the local energy-preserving method is denoted by LEP, and the global energy-preserving method by GEP. For the GEP and LIGEP schemes, two different choices are considered for approximating the spatial derivative: the central difference operator δ^c_x defined by $\delta^c_x v^n_j := \frac{1}{2}(\delta_x v^n_{j-1} + \delta_x v^n_j)$ and the first order Fourier pseudospectral operator [4]. The latter results in the $M \times M$ matrix D, given explicitly by its elements

$$D_{i,j} = \begin{cases} \frac{\pi}{P}(-1)^{i+j} \cot\big(\pi(i-j)/M\big), & \text{if} \quad i \neq j, \\ 0, & \text{if} \quad i = j, \end{cases}$$

evaluated on the domain $[0, P]$, where we assume M even and periodic boundary conditions [12]. If M is odd, we have instead

$$D_{i,j} = \begin{cases} \frac{\pi}{P}(-1)^{i+j} \cot\big(\pi(i-j)/M\big), & \text{if} \quad |i-j| < M/2, \\ \frac{\pi}{P}(-1)^{i+j} \cot\big(\pi(j-i)/M\big), & \text{if} \quad |i-j| > M/2, \\ 0, & \text{if} \quad i = j. \end{cases}$$

7.5.1 Korteweg–de Vries equation

Consider the multi-symplectic structure of the KdV equation as presented in Example 7.1. Applying the LILEP scheme (7.17) to (7.13), we obtain

$$\begin{aligned}\frac{1}{2}\delta_t\mu_x u_j^n + \delta_x\mu_t w_j^n &= 0,\\ -\frac{1}{2}\delta_t\mu_x\phi_j^n - \gamma\delta_x\mu_t v_j^n &= -\mu_t\mu_x w_j^n + \frac{\eta}{2}\mu_x u_j^n\mu_x u_j^{n+1},\\ \gamma\delta_x\mu_t u_j^n &= \mu_t\mu_x v_j^n,\\ \delta_x\mu_t\phi_j^n &= \mu_t\mu_x u_j^n.\end{aligned}$$

By eliminating the auxiliary varibles ϕ, v and w, we see that this is equivalent to

$$\delta_t\mu_t\mu_x^3 u_j^n + \frac{\eta}{2}\delta_x\mu_t\mu_x(\mu_x u_j^n\mu_x u_j^{n+1}) + \gamma^2\delta_x^3\mu_t^2 u_j^n = 0.$$

Omitting the average operator μ_t gives us

$$\delta_t\mu_x^3 u_j^n + \frac{\eta}{2}\delta_x\mu_x(\mu_x u_j^n\mu_x u_j^{n+1}) + \gamma^2\delta_x^3\mu_t u_j^n = 0. \tag{7.37}$$

The polarised discrete energy preserved by this scheme is

$$\bar{\mathcal{E}}_L^n = \Delta x\sum_{j=0}^{M-1}\Big(-\frac{1}{6}\gamma^2\big((\delta_x u_j^n)^2 + 2\delta_x u_j^n\delta_x u_j^{n+1}\big) + \frac{1}{6}\eta\big(\mu_x u_j^n\big)^2\mu_x u_j^{n+1}\Big). \tag{7.38}$$

On the other hand, the discrete energy preserved by the LEP method of [18] is

$$\mathcal{E}_L^n = \Delta x\sum_{j=0}^{M-1}\Big(-\frac{1}{2}\gamma^2(\delta_x u_j^n)^2 + \frac{1}{6}\eta(\mu_x u_j^n)^3\Big). \tag{7.39}$$

By Proposition 7.1 and elimination of the variables ϕ, v and w, (7.38) can be expressed as a modification of (7.39): we may rewrite (7.37) as

$$\begin{aligned}&u_j^{n+1} - u_j^n\\ &= -\Delta t\big(\mu_x^3 + \frac{\Delta t}{2}\gamma^2\delta_x^3 + \frac{\Delta t}{2}\eta\delta_x\mu_x\mathrm{diag}(\mu_x u_n)\mu_x\big)^{-1}\big(\gamma^2\delta_x^3 u^n + \frac{\eta}{2}\delta_x\mu_x(\mu_x u^n)^2\big),\end{aligned}$$

where $(\mu_x u^n)^2$ denotes the element-wise square of $\mu_x u^n$. Inserting this in (7.38), we get

$$\begin{aligned}\bar{\mathcal{E}}_L^n =&\mathcal{E}_L^n - \frac{\Delta t\,\Delta x}{3}\big(-\gamma^2\delta_x^T\delta_x u^n + \frac{\eta}{2}\mu_x^T(\mu_x u^n)^2\big)^T\\ &\big(\mu_x^3 + \frac{\Delta t}{2}\gamma^2\delta_x^3 + \frac{\Delta t}{2}\eta\delta_x\mu_x\mathrm{diag}(\mu_x u^n)\mu_x\big)^{-1}\big(\gamma^2\delta_x^3 u^n + \frac{\eta}{2}\delta_x\mu_x(\mu_x u^n)^2\big)\\ =&\mathcal{E}_L^n + \frac{\Delta t}{3}(\nabla\mathcal{E}_L^n)^T\big(\mu_x^3 - \frac{\Delta t}{2}\zeta_L'(u^n)\big)^{-1}\zeta_L(u^n),\end{aligned}$$

with

$$\zeta_L(u^n) = -\gamma^2\delta_x^3 u^n - \frac{\eta}{2}\delta_x\mu_x(\mu_x u^n)^2,$$

where $\nabla\mathcal{E}_L^n$ means the gradient of $\mathcal{E}_L^n$ with respect to u^n, and $\zeta_L'(u^n)$ denotes the Jacobian matrix of $\zeta_L(u^n)$.

Similarly for the LIGEP method (7.27); applying it to the the multi-symplectic KdV equations (7.13) and eliminating the auxiliary varibles ϕ, v and w, we obtain

$$\delta_t\mu_t u_j^n + \frac{\eta}{2}\mu_t(D(u^n u^{n+1}))_j + \gamma^2\mu_t^2(D^3u^n)_j = 0,$$

where $u^n u^{n+1}$ denotes element-wise multiplication of the vectors. Omitting the average operator μ_t, we get

$$\delta_t u_j^n + \frac{\eta}{2}(D(u^n u^{n+1}))_j + \gamma^2\mu_t(D^3u^n)_j = 0. \tag{7.40}$$

The discrete global energy preserved by the GEP method is

$$\mathcal{E}^n = \Delta x\sum_{j=0}^{M-1}\Big(-\frac{1}{2}\gamma^2(Du^n)_j^2 + \frac{1}{6}\eta(u_j^n)^3\Big), \tag{7.41}$$

while the polarised discrete energy preserved by (7.40) is

$$\begin{aligned}\bar{\mathcal{E}}^n =& \Delta x\sum_{j=0}^{M-1}\Big(-\frac{1}{6}\gamma^2\big((Du^n)_j^2 + 2(Du^n)_j(D^3u^{n+1})_j\big) + \frac{1}{6}\eta\big(u_j^n)^2u_j^{n+1}\Big)\\ =&\mathcal{E}^n - \frac{\Delta t\Delta x}{3}\big(-\gamma^2D^TDu^n + \frac{\eta}{2}(u^n)^2\big)^T\\ &\big(I + \frac{\Delta t}{2}\gamma^2D^3 + \frac{\Delta t}{2}\eta D\,\mathrm{diag}(u^n)\big)^{-1}\big(\gamma^2D^3u^n + \frac{\eta}{2}D(u^n)^2\big)\\ =&\mathcal{E}^n + \frac{\Delta t}{3}(\nabla\mathcal{E}^n)^T\big(I - \frac{\Delta t}{2}\zeta'(u^n)\big)^{-1}\zeta(u^n),\end{aligned} \tag{7.42}$$

where $\zeta(u^n) = -\gamma^2D^3u^n - \frac{\eta}{2}D(u^n)^2$.

Test problem 1

In the first numerical experiment, we consider the problem introduced in [36] and then used by Zhao and Qin [38] and Ascher and McLachlan [1] to test various symplectic and multi-symplectic schemes: the KdV equation with $\gamma = 0.022$, $\eta = 1$, and initial value

$$u_0(x) = \cos(\pi x),$$

with $x \in [0, P]$, $P = 2$. This problem is also considered in Example 3 of [18], where it is solved by implicit schemes that preserve local and/or global energy.

As observed by Gong et al., the global energy-preserving scheme (GEP) with the central difference operator used to approximate ∂_x gives unsatisfactory results for this problem; we observed that the same is true for the LIGEP scheme. Therefore, the Fourier pseudospectral operator is used to approximate the spatial derivatives in the GEP and LIGEP schemes. This seems to result in more accurate solutions than the LEP and LILEP schemes for the same number of discretization points, but at a considerably higher computational cost, as seen from Table 7.1. From Figure 7.1, we can conclude that our linearly implicit schemes give results close to their fully implicit counterparts introduced in [18], and that the different schemes converge to the same solution. Here and in the following test problem, we have solved the fully implicit schemes in each step by Newton's method until $\|F(u^n)\|_2 < 10^{-10}$.

M	200	400	600	1000	1500	2000
LEP	1.87	3.16	4.43	13.81	21.53	28.54
LILEP	4.24e-1	7.40e-1	1.07	1.73	2.67	3.58
GEP	12.29	78.11	242.48	1888.69	5793.18	13154.20
LIGEP	2.16	11.15	33.50	136.93	398.53	894.52

Table 7.1: Computational time, in seconds, for finding the solution of the first test problem at time $t = 5$ by a temporal step size $\Delta t = 0.005$ and various number of discretization points in space, M.

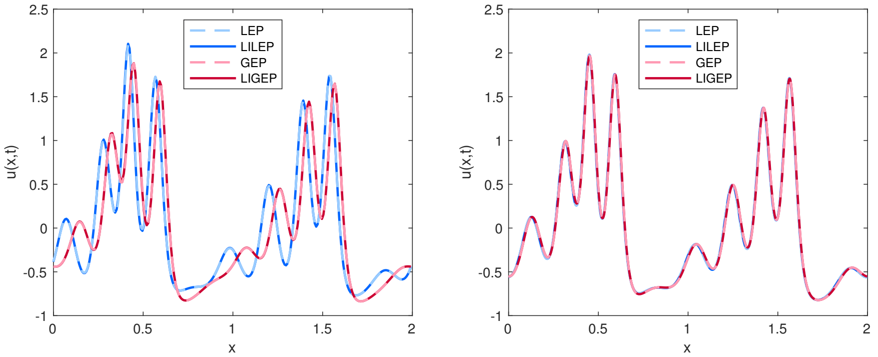


Figure 7.1: Solution of test problem 1 at time $t = 5$ by our schemes and the fully implicit schemes of Gong et al. *Left:* $M = 250$, $\Delta t = 0.02$. *Right:* $M = 1000$, $\Delta t = 0.002$.

Compared to the schemes tested in [1, 38], our schemes do also perform well; see Figure 7.2, where we have plotted solutions by our schemes for the same discretization parameters used in Example 5.3 of [1]. The reference solution is found by the implicit midpoint scheme of [1] with $\theta = 1$ and very fine discretization in space and time: $M = 2000$ and $\Delta t = 0.0001$. We observe that the LILEP scheme behaves similarly to the multi-symplectic box scheme of Arscher and McLachlan (see figures 3 and 4 in [1]), seemingly with the same superior stability for rough discretization in space and time. The LIGEP scheme, on the other hand, starts to blow up at around $t = 1$ when $M = 60$, $\Delta t = 1/150$, but produces for $M = 100$, $\Delta t = 0.004$ a solution that is much closer to the correct solution than any of the schemes tested in [1] (see Figure 3 in that paper for comparison).

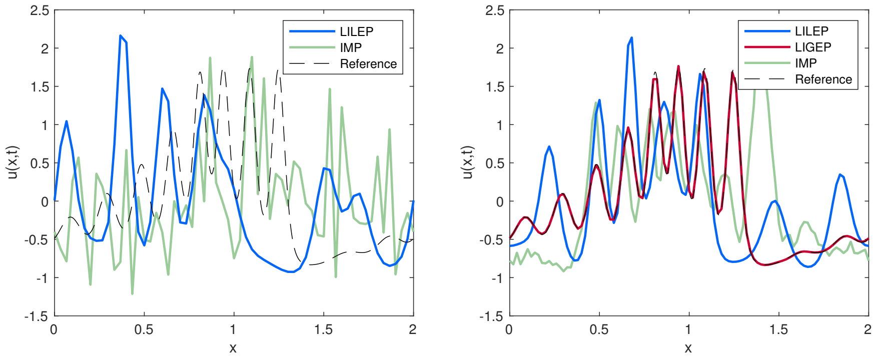


Figure 7.2: Solutions of test problem 1 at time $t = 10$ by our schemes and the implicit midpoint scheme (IMP) as given in [1] (with $\theta = 2/3$ in the left figure and $\theta = 1$ in the right figure). *Left:* $M = 60$, $\Delta t = 1/150$. *Right:* $M = 100$, $\Delta t = 0.004$.

Test problem 2

To get quantitative results on the performance of our methods, we wish to study a problem with a known solution. For the KdV equation with $\gamma = 1$, $\eta = 6$, initial value $u_0(x) = \frac{1}{2}c\,\mathrm{sech}^2(-x + P/2)$ and periodic boundary conditions $u(x+P, t) = u(x, t)$, the exact solution is a soliton moving with a constant speed c in the positive x-direction while keeping its initial shape. That is,

$$u(x,t) = \frac{1}{2}c\,\mathrm{sech}^2((-x+ct) \bmod P - P/2).$$

In our numerical experiments, $c = 4$ and $P = 20$. For this problem, we have used the central difference operator to approximate ∂_x in the GEP and LIGEP schemes, since it gives good results and yields considerably shorter computational time than if the pseudospectral operator is used. The proposed methods all show very good stability conditions when applied to this problem, as expected by methods conserving some invariant. The initial shape of the wave is well kept for long integration times, even when quite large step sizes in space and time are used; Figure 7.3 gives a good illustration of this. As in the previous example, we again observe that little is lost in accuracy by choosing linearly implicit over fully implicit time integration. A close inspection of Figure 7.3 also indicates that the local energy-preserving schemes preserve the shape of the wave better than the global energy-preserving schemes, while on the other hand, the GEP and LIGEP schemes are better than the LEP and LILEP schemes at preserving the speed of the wave. This is confirmed in Table 7.2 by measuring the shape error

$$\epsilon_{\text{shape}} := \min_{\tau} \| U^N - u(\cdot - \tau) \|_2^2$$

and phase error

$$\epsilon_{\text{phase}} := c\,|\operatorname*{argmin}_{\tau} \| U^N - u(\cdot - \tau) \|_2^2 - ct|,$$

where U^N is the numerical solution at end time t.

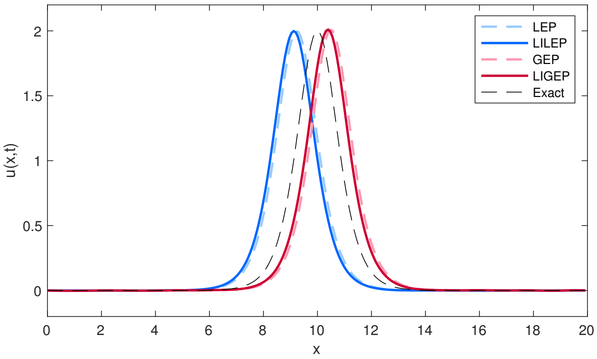


Figure 7.3: The soliton solution of the KdV equation at time $t = 100$, with $M = 250$ discretization points in space and a time step $\Delta t = 0.01$.

M	200			600		
	ϵ_{shape}	ϵ_{phase}	CT	ϵ_{shape}	ϵ_{phase}	CT
LEP	4.67e-3	1.12	21.86	5.86e-4	2.43e-1	51.92
LILEP	4.10e-3	1.23	5.14	1.45e-4	3.50e-1	10.89
GEP	1.62e-2	8.61e-1	19.53	1.71e-3	2.32e-2	49.45
LIGEP	1.71e-2	7.50e-1	6.84	2.47e-3	1.31e-1	12.52

Table 7.2: Phase and shape errors and the computational time (CT) for different schemes applied to test problem 2 of the KdV equation, for varying number of discretization points M, with time step $\Delta t = 0.01$ and end time $t = 100$.

In Figure 7.4, we have plotted the computational time required to reach a certain accuracy in the global error for the different methods, both at time $t = 0.5$ and at time $t = 10$. We compare our methods to the fully implicit LEP and GEP schemes of [18], and also to two of the schemes studied in [1]: the multi-symplectic box scheme (MSB) and the implicit midpoint scheme (IMP). Most notably we see from both plots in Figure 7.4 that the linearly implicit schemes perform better than the fully implicit schemes. Also, we see that at time $t = 0.5$ the global error is lowest for the LILEP scheme, while at $t = 10$ it is lowest for the LIGEP scheme. This is in accordance with the schemes' phase and shape errors, which can be observed from Figure 7.3 and Table 7.2; with increasing time, the phase error becomes more dominant, and thus the scheme with the smallest phase error becomes increasingly advantageous.

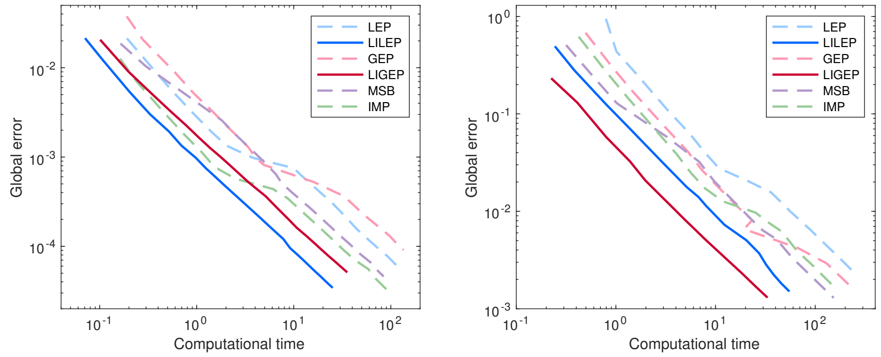


Figure 7.4: Computational time required to reach a given global error, with $\frac{\Delta x}{\Delta t}$ fixed, for test problem 2 of the KdV equation solved at time t. *Left:* $t = 0.5$, $\frac{\Delta x}{\Delta t} = 40$. *Right:* $t = 10$, $\frac{\Delta x}{\Delta t} = 8$.

Figure 7.5 illustrates how the different schemes preserve a discrete approximation to the energy to machine precision. That is, the linearly implicit schemes LILEP and LIGEP preserve exactly the discrete energies (7.38) and (7.42), respectively, while keeping the discrete energies (7.39) and (7.41), respectively, within some bound which depends on the discretization parameters. Likewise, the reverse is true for the fully implicit schemes. These observations fit well with our above results about the different discrete approximations to the energy: that for both the local energy preserving and the global energy preserving schemes, either discrete energy given can be seen as a modification of the other approximation. Finally, we have included plots in Figure 7.6 which confirm that our schemes are of second order in space and time.

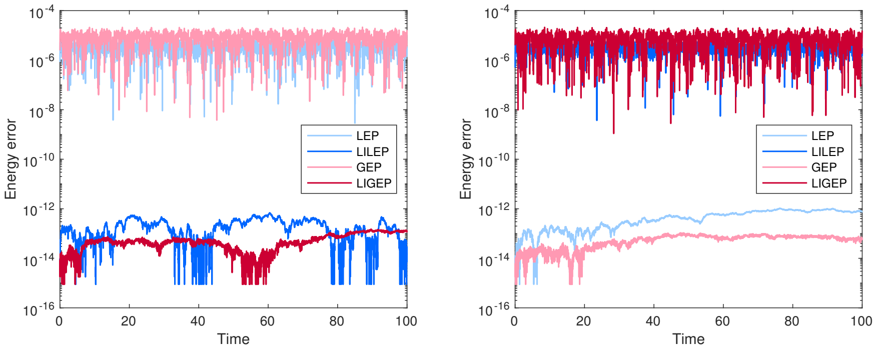


Figure 7.5: Error in discrete approximations to the global energy, by our methods and the fully implicit schemes of Gong et al. *Left:* The error in (7.38) for LEP/LILEP and the error in (7.42) for GEP/LIGEP, for test problem 2 solved with $M = 250$ discretization points in space and time step $\Delta t = 0.01$. *Right:* The error in (7.39) for LEP/LILEP and the error in (7.41) for GEP/LIGEP.

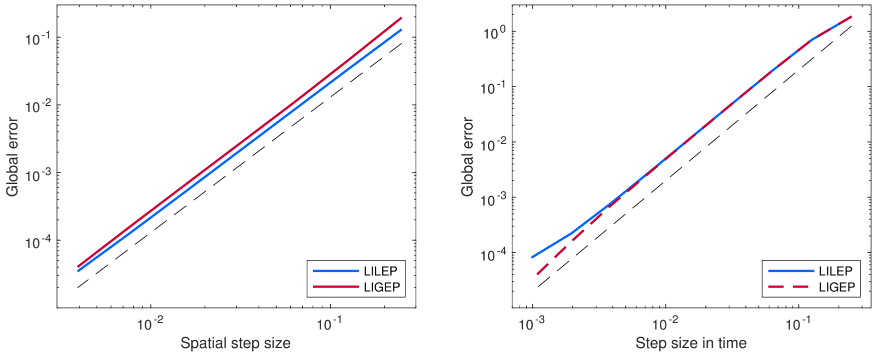


Figure 7.6: Order plots for the LILEP and LIGEP schemes, solving the second test problem for the KdV equation at time $t = 1$. The black, dashed line is a reference line with slope 2 in both plots. *Left:* Fixed temporal step $\Delta t = 2 \times 10^{-4}$. *Right:* Fixed spatial step $\Delta x = 4 \times 10^{-3}$.

7.5.2 Zakharov–Kuznetsov equation

Kahan's method is previously shown to have nice properties when applied to integrable ODE systems [8, 10], and to perform well compared to other linearly implicit methods when applied to the KdV and Camassa–Holm equations [15], which are completely integrable PDEs. We wish to test our methods also on non-integrable systems, as well as on higher-dimensional problems. Therefore we consider the Zakharov–Kuznetsov equation, which is a non-integrable PDE [20, 32]. This two-dimensional generalisation of the KdV equation has a variety of applications, see e.g. [21] for a brief summary.

Applying the (2+1)-dimensional LILEP scheme from Corollary 3 to the Zakharov–Kuznetsov equation (7.14) multi-symplectified as described in Example 7.2, we find

$$\begin{aligned}
\delta_x\mu_t\mu_y\phi^n_{j,k} &= \mu_t\mu_x\mu_y u^n_{j,k},\\
\frac{1}{2}\delta_t\mu_x\mu_y\phi^n_{j,k}+\delta_x\mu_t\mu_y v^n_{j,k}+\delta_y\mu_t\mu_x w^n_{j,k} &= \mu_t\mu_x\mu_y p^n_{j,k}-\frac{1}{2}\mu_x\mu_y u^n_{j,k}\mu_x\mu_y u^{n+1}_{j,k},\\
\delta_x\mu_t\mu_y w^n_{j,k}-\delta_y\mu_t\mu_x v^n_{j,k} &= 0,\\
-\frac{1}{2}\delta_t\mu_x\mu_y u^n_{j,k}-\delta_x\mu_t\mu_y p^n_{j,k} &= 0,\\
-\delta_x\mu_t\mu_y u^n_{j,k}+\delta_y\mu_t\mu_x q^n_{j,k} &= -\mu_t\mu_x\mu_y v^n_{j,k},\\
-\delta_x\mu_t\mu_y q^n_{j,k}-\delta_y\mu_t\mu_x u^n_{j,k} &= -\mu_t\mu_x\mu_y w^n_{j,k}.
\end{aligned}$$

Upon eliminating all variables except u, we are left with

$$\begin{aligned}
\delta_t\mu_t\mu_x^3\mu_y u^n_{j,k} &+ \frac{1}{2}\delta_x\mu_t\mu_x\mu_y(\mu_x\mu_y u^n_{j,k}\mu_x\mu_y u^{n+1}_{j,k})\\
&+\delta_x^3\mu_t^2\mu_y^2 u^n_{j,k}+\delta_x\delta_y^2\mu_t^2\mu_x^2 u^n_{j,k}=0.
\end{aligned}$$

The operator μ_t is again superfluous. Hence we get the scheme

$$\delta_t\mu_x^3\mu_y u^n_{j,k}+\frac{1}{2}\delta_x\mu_x\mu_y(\mu_x\mu_y u^n_{j,k}\mu_x\mu_y u^{n+1}_{j,k})+\delta_x^3\mu_t\mu_y^2 u^n_{j,k}+\delta_x\delta_y^2\mu_t\mu_x^2 u^n_{j,k}=0.$$

This scheme preserves

$$\begin{aligned}
\bar{\mathcal{E}}^n_L = \frac{1}{6}\Delta x\Delta y\sum_{j=0}^{M_x-1}\sum_{k=0}^{M_y-1}&\Big(2\delta_x\mu_y u^{n+1}_{j,k}\delta_x\mu_y u^n_{j,k}+(\delta_x\mu_y u^n_{j,k})^2\\
&+2\delta_y\mu_x u^{n+1}_{j,k}\delta_y\mu_x u^n_{j,k}+(\delta_y\mu_x u^n_{j,k})^2-(\mu_x\mu_y u^n_{j,k})^2(\mu_x\mu_y u^{n+1}_{j,k})\Big),
\end{aligned}$$

which is a two-step discrete approximation of the energy

$$\mathcal{E}=\int(\frac{1}{2}(\nabla u)^2-\frac{1}{6}u^3)\,d\Omega.$$

Similarly, applying the linearly implicit global energy-preserving method (7.36) to (7.16), we get the scheme

$$\delta_t u^n_{j,k}+\frac{1}{2}(D_x(u^n u^{n+1}))_{j,k}+\mu_t(D_x^3(u^n))_{j,k}+\mu_t(D_xD_y^2(u^n))_{j,k}=0,$$

which preserves the two-step discrete energy approximation

$$\begin{aligned}
\bar{\mathcal{E}}^n = \frac{1}{6}\Delta x\Delta y\sum_{j=0}^{M_x-1}\sum_{k=0}^{M_y-1}&\Big(2(D_x u^n)_{j,k}(D_x u^{n+1})_{j,k}+((D_x u^n)_{j,k})^2\\
&+2(D_y u^n)_{j,k}(D_y u^{n+1})_{j,k}+((D_y u^n)_{j,k})^2-(u^n_{j,k})^2 u^{n+1}_{j,k}\Big).
\end{aligned}$$

Test problem

Taking a note from a numerical experiment performed in [4], we study the formation of cylindrical soliton pulses on the domain $[0,P] \times [0,P]$, $P = 30$, following the initial condition

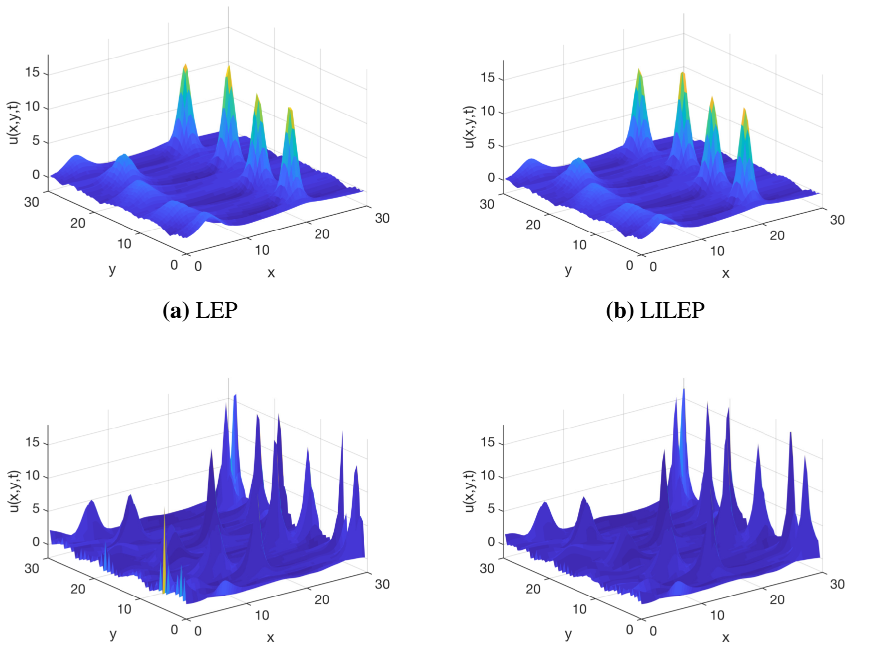
$$u_0(x,y) = 3c\,\mathrm{sech}^2\big(\frac{1}{2}\sqrt{c}(x - P/2)\big) + \xi(y),$$

where $\xi(y)$ is a random perturbation.

Upon trying the different schemes we can immediately conclude that the local energy-preserving schemes are superior for this problem when compared to the global energy-preserving schemes. The GEP and LIGEP schemes are too costly when the pseudospectral operator is used, and gives oscillatory behaviour in the y-direction when the central difference operator is used, unless the discretization in this direction is very fine. Although the global energy-preserving schemes with the central difference operator are slightly faster then the local energy-preserving schemes, as can be seen in Table 7.3, this is undermined by the cost of the extra discretization points needed to avoid oscillations in the former case. As was the case for the KdV problem, we see little difference between the linearly implicit schemes and their fully implicit counterparts. This can be seen in Figure 7.7, as can the oscillations in y-direction of the solution found by the GEP and LIGEP methods. The plots in Figure 7.7 can be compared to the plot in Figure 7.8, where the same problem is solved by the LILEP method using finer discretization in space and time. The initial random perturbation in y-direction over 75 points is then transferred over to 225 points using linear interpolation.

M	45	75	105	135	165	195	225
LEP	5.10	32.20	48.43	101.59	125.23	258.64	353.98
LILEP	2.04	8.87	14.57	31.02	37.25	78.98	108.02
GEP	3.62	19.54	41.87	73.59	122.31	186.74	258.19
LIGEP	1.38	6.00	13.45	23.79	39.31	60.27	83.32

Table 7.3: Running time, in seconds, for computing 100 steps in time by the various schemes and various number of discretization points $M = M_x = M_y$ in each spatial direction, solving our test problem for the Zakharov–Kuznetsov equation. As for the KdV equation test problems, a tolerance of 10^{-10} is used when solving the fully implicit schemes by Newton's method.



(a) LEP

(b) LILEP

(c) GEP

(d) LIGEP

Figure 7.7: The test problem of the Zakharov–Kuznetsov equation solved at time $t = 15$ by the different schemes, with $M = M_x = M_y = 75$ points in each spatial direction and $\Delta t = 0.1$.

7.6 Concluding remarks

In this paper, we propose two types of linearly implicit methods with conservation properties for cubic invariants of multi-symplectic PDEs. The linearly implicit local energy-preserving (LILEP) method preserves a discrete approximation to the local energy conservation law, and by extension, the global energy whenever periodic boundary conditions are considered. The linearly implicit global energy-preserving (LIGEP) method preserves the global energy without inheriting the local preservation from the continuous system.

We test our methods on two PDEs: the one-dimensional, integrable Korteweg–de Vries (KdV) equation and the two-dimensional, non-integrable Zakharov–Kuznetsov equation. The numerical experiments confirm that the proposed methods are of second order both in space and time and that they preserve the expected local and global energy conservation laws. We have observed excellent stability properties for the LILEP scheme in particular, and very high accuracy in the LIGEP scheme even for quite coarse discretization when a

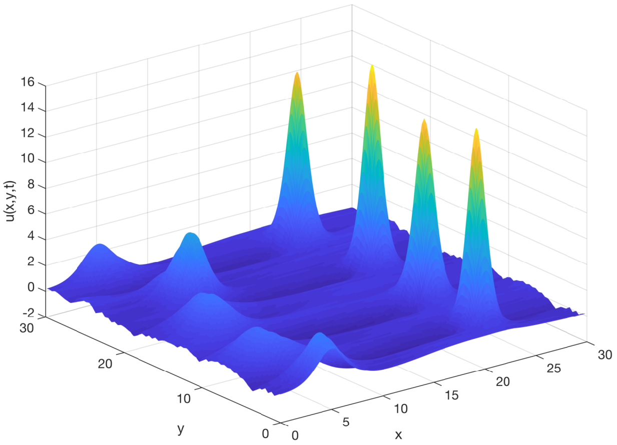


Figure 7.8: The test problem of the Zakharov–Kuznetsov equation solved at time $t = 15$ by the LILEP scheme, with $M = M_x = M_y = 225$ discretization points in each spatial direction and a temporal step size $\Delta t = 0.001$.

Fourier pseudospectral operator is used to approximate the spatial derivative. Compared to the fully implicit methods of Gong et al. in [18], which was an inspiration for this paper, our methods show comparable wave profiles, global errors and energy errors, at a significantly lower computational cost. For two-dimensional problems, where fully implicit schemes quickly become very expensive to compute, the combination of local energy-preservation and a linearly implicit method seems to provide for a very competitive method.

Although we have only considered the preservation of cubic invariants in this paper, our schemes can be extended to preserve higher order polynomials by the polarisation techniques for generalising Kahan's method suggested in [9]. This would result in $(p-2)$-step methods for preservation of a discrete p-order polynomial invariant.

Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 691070. The authors wish to express gratitude to Elena Celledoni and Brynjulf Owren for constructive discussions and helpful suggestions during our work on this paper, and to Benjamin Tapley for helping with the language.

Bibliography

[1] U. M. Ascher and R. I. McLachlan. On symplectic and multisymplectic schemes for the KdV equation. *J. Sci. Comput.*, 25(1-2):83–104, 2005.

[2] 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(1962):1365–1395, 1997.

[3] T. J. Bridges. Multi-symplectic structures and wave propagation. volume 121, pages 147–190, 1997.

[4] T. J. Bridges and S. Reich. Multi-symplectic spectral discretizations for the Zakharov–Kuznetsov and shallow water equations. *Phys. D*, 152/153:491–504, 2001. Advances in nonlinear mathematics and science.

[5] L. Brugnano, F. Iavernaro, and D. Trigiante. Hamiltonian boundary value methods (energy preserving discrete line integral methods). *JNAIAM. J. Numer. Anal. Ind. Appl. Math.*, 5(1-2):17–37, 2010.

[6] W. Cai, H. Li, and Y. Wang. Partitioned averaged vector field methods. *J. Comput. Phys.*, 370:25–42, 2018.

[7] 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(20):6770–6789, 2012.

[8] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, and G. R. W. Quispel. Integrability properties of Kahan's method. *J. Phys. A*, 47(36):365202, 20, 2014.

[9] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, and G. R. W. Quispel. Discretization of polynomial vector fields by polarization. *Proc. A.*, 471(2184):20150390, 10, 2015.

[10] E. Celledoni, R. I. McLachlan, B. Owren, and G. R. W. Quispel. Geometric properties of Kahan's method. *J. Phys. A*, 46(2):025201, 12, 2013.

[11] E. Celledoni, D. I. McLaren, B. Owren, and G. R. W. Quispel. Geometric and integrability properties of Kahan's method: the preservation of certain quadratic integrals. *J. Phys. A*, 52(6):065201, 9, 2019.

[12] Y. Chen, S. Song, and H. Zhu. The multi-symplectic Fourier pseudospectral method for solving two-dimensional Hamiltonian PDEs. *J. Comput. Appl. Math.*, 236(6):1354–1369, 2011.

[13] S. H. Christiansen, H. Z. Munthe-Kaas, and B. Owren. Topics in structure-preserving discretization. *Acta Numer.*, 20:1–119, 2011.

[14] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33(5):2318–2340, 2011.

[15] S. Eidnes, L. Li, and S. Sato. Linearly implicit structure-preserving schemes for Hamiltonian systems. *arXiv preprint, arXiv:1901.03573*, 2019.

[16] Z. Fei, V. M. Pérez-García, and L. Vázquez. Numerical simulation of nonlinear Schrödinger systems: a new conservative scheme. *Appl. Math. Comput.*, 71(2-3):165–177, 1995.

[17] D. Furihata and T. Matsuo. *Discrete variational derivative method*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing. CRC Press, Boca Raton, FL, 2011. A structure-preserving numerical method for partial differential equations.

[18] Y. Gong, J. Cai, and Y. Wang. Some new structure-preserving algorithms for general multi-symplectic formulations of Hamiltonian PDEs. *J. Comput. Phys.*, 279:80–102, 2014.

[19] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[20] H.-C. Hu. New exact solutions of Zakharov–Kuznetsov equation. *Commun. Theor. Phys. (Beijing)*, 49(3):559–561, 2008.

[21] H. Iwasaki, S. Toh, and T. Kawahara. Cylindrical quasi-solitons of the Zakharov–Kuznetsov equation. *Phys. D*, 43(2-3):293–303, 1990.

[22] C. Jiang, W. Cai, and Y. Wang. A linear-implicit and local energy-preserving scheme for the sine-Gordon equation based on the invariant energy quadratization approach. *arXiv preprint, arXiv:1808.06854*, 2018.

[23] C. Jiang, Y. Gong, W. Cai, and Y. Wang. A linearly implicit structure-preserving scheme for the Camassa–Holm equation based on multiple

scalar auxiliary variables approach. *arXiv preprint, arXiv:1907.00167*, 2019.

[24] W. Kahan. Unconventional numerical methods for trajectory calculations. *Unpublished lecture notes*, 1:13, 1993.

[25] R. A. LaBudde and D. Greenspan. Energy and momentum conserving methods of arbitrary order for the numerical integration of equations of motion. II. Motion of a system of particles. *Numer. Math.*, 26(1):1–16, 1976.

[26] B. Leimkuhler and S. Reich. *Simulating Hamiltonian dynamics*, volume 14 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2004.

[27] Y.-W. Li and X. Wu. General local energy-preserving integrators for solving multi-symplectic Hamiltonian PDEs. *J. Comput. Phys.*, 301:141–166, 2015.

[28] J. E. Marsden, G. W. Patrick, and S. Shkoller. Multisymplectic geometry, variational integrators, and nonlinear PDEs. *Comm. Math. Phys.*, 199(2):351–395, 1998.

[29] T. Matsuo and D. Furihata. Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations. *J. Comput. Phys.*, 171(2):425–447, 2001.

[30] R. I. McLachlan, G. R. W. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *R. Soc. Lond. Philos. Trans. Ser. A Math. Phys. Eng. Sci.*, 357(1754):1021–1045, 1999.

[31] B. E. Moore and S. Reich. Multi-symplectic integration methods for Hamiltonian PDEs. *Future Generation Computer Systems*, 19(3):395–402, 2003.

[32] H. Nishiyama, T. Noi, and S. Oharu. Conservative finite difference schemes for the generalized Zakharov–Kuznetsov equations. *J. Comput. Appl. Math.*, 236(12):2998–3006, 2012.

[33] S. Reich. Multi-symplectic Runge-Kutta collocation methods for Hamiltonian wave equations. *J. Comput. Phys.*, 157(2):473–499, 2000.

[34] Y. Wang, B. Wang, and M. Qin. Local structure-preserving algorithms for partial differential equations. *Sci. China Ser. A*, 51(11):2115–2136, 2008.

[35] X. Yang, J. Zhao, and Q. Wang. Numerical approximations for the molecular beam epitaxial growth model based on the invariant energy quadratization method. *J. Comput. Phys.*, 333:104–127, 2017.

[36] N. J. Zabusky and M. D. Kruskal. Interaction of "solitons" in a collisionless plasma and the recurrence of initial states. *Phys. Rev. Lett.*, 15(6):240, 1965.

[37] V. Zakharov and E. Kuznetsov. Three-dimensional solitons. *Zh. Eksp. Teor. Fiz*, 66:594–597, 1974.

[38] P. F. Zhao and M. Z. Qin. Multisymplectic geometry and multisymplectic Preissmann scheme for the KdV equation. *J. Phys. A*, 33(18):3613–3626, 2000.

Shape analysis on Lie groups and homogeneous spaces

Elena Celledoni, Sølve Eidnes, Markus Eslitzbichler and Alexander Schmeding

Published in Proceedings for Geometric Science of Information 2017,

Lecture Notes in Computer Science 10589, pp. 49–56 (2017)

Shape analysis on Lie groups and homogeneous spaces

Abstract. In this paper we are concerned with the approach to shape analysis based on the so called Square Root Velocity Transform (SRVT). We propose a generalisation of the SRVT from Euclidean spaces to shape spaces of curves on Lie groups and on homogeneous manifolds. The main idea behind our approach is to exploit the geometry of the natural Lie group actions on these spaces.

Shape analysis methods have significantly increased in popularity in the last decade. Advances in this field have been made both in the theoretical foundations and in the extension of the methods to new areas of application. Originally developed for planar curves, the techniques of shape analysis have been successfully extended to higher dimensional curves, surfaces, activities, character motions and a number of different types of digitalized objects.

In the present paper, shapes are unparametrized curves, evolving on a vector space, on a Lie group, or on a manifold. Shape spaces and spaces of curves are infinite-dimensional Riemannian manifolds, whose Riemannian metrics are the crucial tool to compare and analyse shapes.

We are concerned with one particular approach to shape analysis, which is based on the Square Root Velocity Transform (SRVT) [10]. On vector spaces, the SRVT maps parametrized curves (i.e. smooth immersions) to appropriately scaled tangent vector fields along them via

$$\mathcal{R}\colon \mathrm{Imm}([0,1],\mathcal{R}^d) \to C^\infty([0,1],\mathcal{R}^d \setminus \{0\}), \quad c \mapsto \frac{\dot{c}}{\sqrt{\|\dot{c}\|}}. \tag{8.1}$$

The transformed curves are then compared computing geodesics in the L^2 metric, and the scaling induces reparametrization invariance of the pullback metric. Note that it is quite natural to consider an L^2 metric directly on the original parametrized curves. Constructing the L^2 metric with respect to integration by arc-length, one obtains a reparametrisation invariant metric. However, this metric is unsuitable for our purpose as it leads to vanishing geodesic distance on the quotient shape space [6] and consequently also on the space of parametrised curves [1]. This infinite-dimensional phenomenon prompted the investigation of alternative, higher order Sobolev type metrics [7], which however can be computationally demanding. Since it allows geodesic computations via the L^2 metric on the transformed curves, the SRVT technique is computationally attractive. It is also possible to prove that this algorithmic approach corresponds, at least locally, to a particular Sobolev type metric, see [2,4].

We propose a generalisation of the SRVT to construct well-behaved Riemannian metrics on shape spaces with values in Lie groups and homogeneous manifolds. Our methodology is alternative to what was earlier proposed in [5, 11] and the main idea is, following [4], to take advantage of the Lie group acting transitively on the homogeneous manifold. Since we want to compare curves, the main tool here is an SRVT which transports the manifold valued curves into the Lie algebra or a subspace of the Lie algebra.

8.1 SRVT for Lie group valued shape spaces

In the Lie group case, the obvious choice for this tangent space is of course the Lie algebra $\mathfrak{g}$ of the Lie group G. The idea is to use the derivative T_eR_g of the right translation for the transport and measure with respect to a right-invariant Riemannian metric.[1] Instead of the ordinary derivative, one thus works with the right-logarithmic derivative $\delta^r(c)(t) = T_eR_{c(t)^{-1}}(\dot{c}(t))$ (here e is the identity element of G) and defines an SRVT for Lie group valued curves as (see [4]):

$$\mathcal{R}\colon \mathrm{Imm}([0,1],G) \to C^\infty([0,1],\mathfrak{g}\setminus\{0\}), \quad c \mapsto \frac{\delta^r(c)}{\sqrt{\|\dot{c}\|}}. \tag{8.2}$$

We will use the short notetion $I = [0,1]$ in what follows. Using tools from Lie theory, we are then able to describe the resulting pullback metric on the space $\mathcal{P}_*$ of immersions $c\colon [0,1] \to G$ which satisfy $c(0) = e$:

Theorem 8.1 (The Elastic metric on Lie group valued shape spaces [4])**.** Let $c \in \mathcal{P}_*$ and consider $v, w \in T_c\mathcal{P}_*$. The pullback of the L^2-metric on $C^\infty(I,\mathfrak{g}\setminus\{0\})$ under the SRVT (8.2) to $\mathcal{P}_*$ is given by the first order Sobolev metric:

$$\begin{aligned} G_c(v,w) = \int_I \frac{1}{4}\left\langle D_s v, u_c\right\rangle\left\langle D_s w, u_c\right\rangle \\ + \Big\langle D_s v - u_c\left\langle D_s v, u_c\right\rangle, D_s w - u_c\left\langle D_s w, u_c\right\rangle\Big\rangle \,\mathrm{d}s, \end{aligned} \tag{8.3}$$

where $D_s v := T_c\delta^r(v)/\|\dot{c}\|$, $u_c := \delta^r(c)/\left\|\delta^r(c)\right\|$ is the unit tangent vector of $\delta^r(c)$ and $\mathrm{d}s = \left\|\dot{c}(t)\right\|\mathrm{d}t$.

The geodesic distance of this metric descends to a nonvanishing metric on the space of unparametrized curves. In particular, this distance is easy to compute as one can prove [4, Theorem 3.16] that

Theorem 8.2. If dim $\mathfrak{g} > 2$, then the geodesic distance of $C^\infty(I,\mathfrak{g}\setminus\{0\})$ is globally given by the L^2-distance. In particular, in this case the geodesic distance of the pullback metric (8.3) on $\mathcal{P}_*$ is given by

$$d_{\mathcal{P}_*}(c_0,c_1) := \sqrt{\int_I \|\mathcal{R}(c_0)(t) - \mathcal{R}(c_1)(t)\|^2\,\mathrm{d}t}.$$

[1] Equivalently one could instead use left translations and a left-invariant metric here.

These tools give rise to algorithms which can be used in, among other things, tasks related to computer animation and blending of curves, as shown in [4]. The blending $c(t,s)$ of two curves $c_0(t)$ and $c_1(t)$, $t \in I$, amounts simply to a convex linear convex combination of their SRV transforms:

$$c(t,s) = \mathcal{R}^{-1}\big(s\mathcal{R}(c_0(t)) + (1-s)\mathcal{R}(c_1(t))\big), \qquad s \in [0,1].$$

Using the transformation of the curves to the Lie algebra by the SRVT, we also propose a curve closing algorithm allowing one to remove discontinuities from motion capturing data while preserving the general structure of the movement. (See Figure 8.1.)

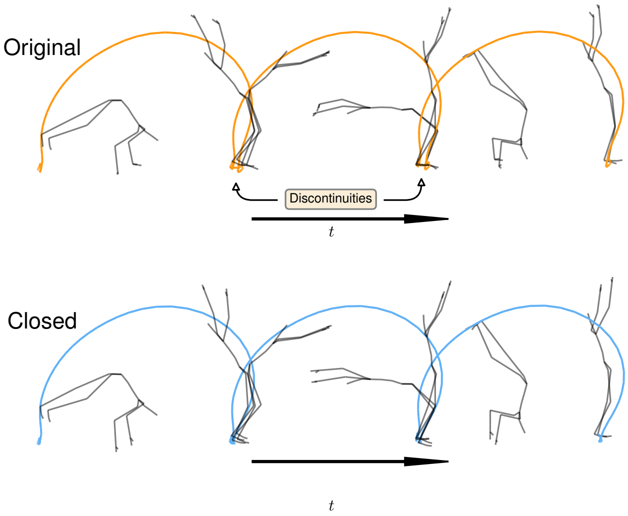


Figure 8.1: Application of closing algorithm to a cartwheel animation. Note the large difference between start and end poses, on the right and the left respectively. The motion is repeated once and suffers from a strong jerk when it repeats, especially in the left hand. In the second row, the curve closing method has been used to alleviate this discontinuity.

8.2 The structure of the SRVT

Analysing the constructions for the square root velocity transform, e.g. (8.1) and (8.2) or the generalisations proposed in the literature, every SRVT is composed of three distinct building blocks. While two of these blocks can not be changed, there are many choices for the second one (transport) in constructing an SRVT:

- **Differentiation**: The basic building block of every SRVT, taking a curve to its derivative.

- **Transport**: Bringing a curve into a common space of reference. In general there are many choices for this transport[2] (in our approach we use the Lie group action to transport data into the Lie algebra of the acting group).

- **Scaling**: The second basic building block, assures reparametrization invariance of the metrics obtained.

In constructing the SRVT, we advocate the use of Lie group actions for the transport. This action allows us to transport derivatives of curves to our choice of base point and to lift this information to a curve in the Lie algebra.

Other common choices for the transport usually arise from parallel transport (cf. e.g. [5, 11]). The advantage of using the Lie group action is that we obtain a global transport, i.e. we do not need to restrict to certain open submanifolds to make sense of the (parallel) transport.[3] Last but not least, right translation is in general computationally more efficient than computing parallel transport using the original Riemannian metric on the manifold.

8.3 SRVT on homogeneous spaces

Our approach [3] for shape analysis on a homogeneous manifold $\mathcal{M} = G/H$ exploits again the geometry induced by the canonical group action $\Lambda\colon G \times \mathcal{M} \to \mathcal{M}$. We fix a Riemannian metric on G which is right H-invariant, i.e. the maps R_h for $h \in H$ are Riemannian isometries. The SRVT is obtained using a right inverse of the composition of the Lie group action with the evolution operator (i.e. the inverse of the right-logarithmic derivative) of the Lie group. If the homogeneous manifold is reductive,[4] there is an explicit way to construct this right inverse. Identifying the tangent space at $[e]$, the equivalence class of the identity, via $\omega_e\colon T_{[e]}\mathcal{M} \to \mathfrak{m} \subseteq \mathfrak{g}$ with the reductive complement. Then we define the map $\omega([g]) = \mathrm{Ad}(g).\omega_e(T\Lambda(g^{-1}, \cdot)[g])$ (which is well-defined by

[2] In the literature, e.g. [11], a common choice is parallel transport with respect to the Riemannian structure.

[3] The problem in these approaches arises from choosing curves along which the parallel transport is conducted. Typically, one wants to transport along geodesics to a reference point and this is only well-defined outside of the cut locus (also cf. [8]).

[4] Recall that a homogeneous space G/H is reductive if the Lie subalgebra $\mathfrak{h}$ of $H \subseteq G$ admits a reductive complement, i.e. $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, where $\mathfrak{m}$ is a subvector space invariant under the adjoint action of H.

reductivity) and obtain a square root velocity transform for reductive homogeneous spaces as

$$\mathcal{R}\colon \operatorname{Imm}([0,1],\mathcal{M}) \to C^{\infty}([0,1],\mathfrak{g}\setminus\{0\}), \quad c \mapsto \frac{\omega\circ\dot{c}}{\sqrt{\|\omega\circ\dot{c}\|}} \tag{8.4}$$

Conceptually this SRVT is somewhat different from the one for Lie groups, as it does not establish a bijection between the manifolds of smooth mappings. However, one can still use (8.4) to construct a pullback metric on the manifold of curves to the homogeneous space by pulling back the L^2 inner product of curves on the Lie algebra through the SRVT. Different choices of Lie group actions will give rise to different Riemannian metrics (with different properties).

8.4 Numerical experiments

We present some results about the realisation of this metric through the SRVT framework in the case of reductive homogeneous spaces. Further, our results are illustrated in a concrete example. We compare the new methods for curves into the sphere SO(3)/SO(2) with results derived from the Lie group case.

In the following, we use the Rodrigues' formula for the Lie group exponential $\exp\colon \mathfrak{so}(3) \to \mathrm{SO}(3)$,

$$\exp(\hat{x}) = I + \frac{\sin(\alpha)}{\alpha}\hat{x} + \frac{1-\cos(\alpha)}{\alpha^2}\hat{x}^2, \qquad \alpha = \|x\|_2$$

and the corresponding formula for the logarithm $\log\colon \mathrm{SO}(3) \to \mathfrak{so}(3)$,

$$\log(X) = \frac{\sin^{-1}(\|y\|)}{\|y\|}\hat{y}, \quad X \neq I, \quad X \text{ close to } I,$$

are used, where $\hat{y} = \frac{1}{2}(X - X^{\mathrm{T}})$, and the relationship between x and $\hat{x}$ is given by the isomorphism between $\mathbb{R}^3$ and $\mathfrak{so}(3)$ known as the hat map

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \hat{x} = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix}.$$

8.4.1 Lie group case

Consider a continuous curve $z(t), t \in [t_0, t_N]$, in SO(3). We approximate it by $\bar{z}(t)$, interpolating between $N+1$ values $\bar{z}_i = z(t_i)$, with $t_0 < t_1 < ... < t_N$, as:

$$\bar{z}(t) := \sum_{i=0}^{N-1} \chi_{[t_i,t_{i+1})}(t) \exp\left(\frac{t-t_i}{t_{i+1}-t_i}\log\left(\bar{z}_{i+1}\bar{z}_i^{\mathrm{T}}\right)\right)\bar{z}_i, \tag{8.5}$$

where χ is the characteristic function.

The SRVT (8.2) of $\bar{z}(t)$ is a piecewise constant function $\bar{p}(t)$ in $\mathfrak{so}(3)$ with values $\bar{p}_i = \bar{p}(t_i)$, $i = 0,...,N-1$, found by

$$\bar{p}_i = \frac{\eta_i}{\sqrt{\|\eta_i\|}}, \qquad \eta_i = \frac{\log(\bar{z}_{i+1}\bar{z}_i^{\mathrm{T}})}{t_{i+1}-t_i}.$$

The inverse $\mathcal{R}^{-1} : \mathfrak{so}(3) \to \mathrm{SO}(3)$ is then given by (8.5), with the discrete points

$$\bar{z}_{i+1} = \exp\left(\|\bar{p}_i\|\bar{p}_i\right)\bar{z}_i, \quad i = 1,...,N-1, \quad \bar{z}_0 = z(t_0).$$

8.4.2 Homogeneous manifold case

As an example of the homogeneous space case, consider the curve $c(t)$ on the sphere SO(3)/SO(2) (i.e. S^2), which we approximate by $\bar{c}(t)$, interpolating between the $N+1$ values $\bar{c}_i = c(t_i)$:

$$\bar{c}(t) := \sum_{i=0}^{N-1} \chi_{[t_i,t_{i+1})}(t) \exp\left(\frac{t-t_i}{t_{i+1}-t_i}\left(v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}\right)\right)\bar{c}_i, \tag{8.6}$$

where v_i are approximations to $\left.\frac{d}{dt}\right|_{t=t_i} c(t)$ found by solving the equations

$$\bar{c}_{i+1} = \exp\left(v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}\right)\bar{c}_i, \tag{8.7}$$

$$\text{constrained by} \quad v_i^{\mathrm{T}}\bar{c}_i = 0. \tag{8.8}$$

Observing that if $\kappa = \bar{c}_i \times v_i$, then $\hat{\kappa} = v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}$, and assuming that the sphere has radius 1, we have by (8.8) that $\|\bar{c}_i \times v_i\|_2 = \|\bar{c}_i\|_2\|v_i\|_2 = \|v_i\|_2$. By (8.7) we get

$$\bar{c}_{i+1} = \frac{\sin(\|v_i\|_2)}{\|v_i\|_2}v_i + \cos\left(\|v_i\|_2\right)\bar{c}_i.$$

Calculations give $\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1} = 1-\cos\left(\|v_i\|_2\right)$ and $\|v_i\|_2 = \arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)$, lead to $v_i = \left(\bar{c}_{i+1} - \bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\bar{c}_i\right)\frac{\arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)}{\sqrt{1-\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)^2}}$ which we insert into (8.6) to get

$$\bar{c}(t) = \sum_{i=0}^{N-1} \chi_{[t_i,t_{i+1})}(t) \exp\left(\frac{t-t_i}{t_{i+1}-t_i}\frac{\arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)}{\sqrt{1-\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)^2}}\left(\bar{c}_{i+1}\bar{c}_i^{\mathrm{T}} - \bar{c}_i\bar{c}_{i+1}^{\mathrm{T}}\right)\right)\bar{c}_i. \tag{8.9}$$

The SRVT (8.4) of $\bar{c}(t)$ is a piecewise constant function $\bar{q}(t)$ in $\mathfrak{so}(3)$, taking values $\bar{q}_i = \bar{q}(t_i)$, $i = 0, ..., N-1$, where

$$\begin{aligned}\bar{q}_i = \mathcal{R}(\bar{c}_i) &= \frac{a_{\bar{c}_i}(v_i)}{\|a_{\bar{c}_i}(v_i)\|^{\frac{1}{2}}} = \frac{v_i \bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}}{\|v_i \bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}\|^{\frac{1}{2}}} \\ &= \frac{\arccos^{\frac{1}{2}}\left(\bar{c}_i^{\mathrm{T}} \bar{c}_{i+1}\right)}{\left(1-\left(\bar{c}_i^{\mathrm{T}} \bar{c}_{i+1}\right)^2\right)^{\frac{1}{4}} \|\bar{c}_{i+1}\bar{c}_i^{\mathrm{T}} - \bar{c}_i \bar{c}_{i+1}^{\mathrm{T}}\|^{\frac{1}{2}}} \left(\bar{c}_{i+1}\bar{c}_i^{\mathrm{T}} - \bar{c}_i \bar{c}_{i+1}^{\mathrm{T}}\right)\end{aligned}$$

The inverse of this SRVT is given by (8.9), with the discrete points found as in the Lie group case by $\bar{c}_{i+1} = \exp\left(\|\bar{q}_i\|\bar{q}_i\right)\bar{c}_i$ and $\bar{c}_0 = c(t_0)$.

In Figure 8.2 we show instants of the computed geodesic in the shape space of curves on the sphere between two curves $\bar{c}_1$ and $\bar{c}_2$. We compare this to the geodesic between the curves $\bar{z}_1$ and $\bar{z}_2$ in SO(3) which when mapped to S^2 gives $\bar{c}_1$ and $\bar{c}_2$. We show the results obtained before and after reparametrization. In the latter case, a dynamic programming algorithm, see [9], was used to reparametrize the curve $\bar{c}_2(t)$ such that its distance to $\bar{c}_1(t)$, measured by taking the L^2 norm of $\bar{q}_1(t) - \bar{q}_2(t)$ in the Lie algebra, is minimized. The various instances of the geodesics between $\bar{c}_1(t)$ and $\bar{c}_2(t)$ are found by interpolation,

$$\bar{c}_{\mathrm{int}}(\bar{c}_1, \bar{c}_2, \theta) = \mathcal{R}^{-1}\left(\left(1-\theta\right)\mathcal{R}(\bar{c}_1) + \theta\,\mathcal{R}(\bar{c}_2)\right), \qquad \theta \in [0,1].$$

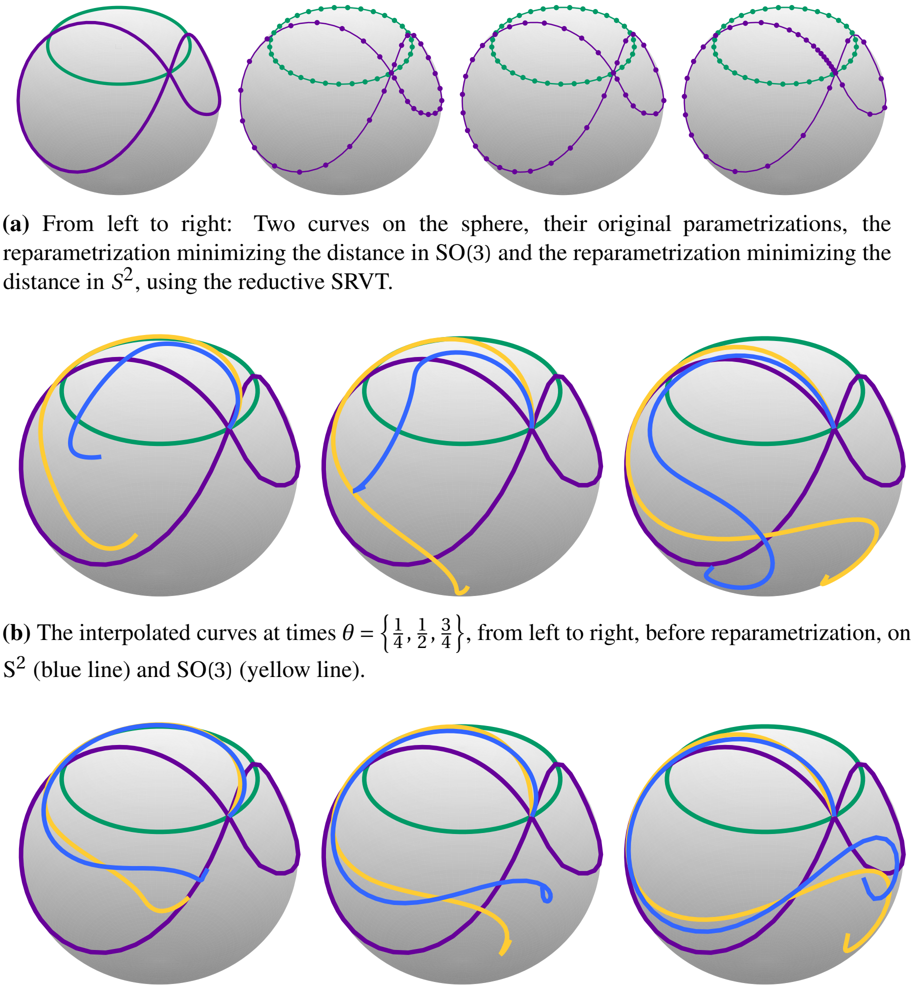
Bibliography

[1] M. Bauer, M. Bruveris, P. Harms, and P. W. Michor. Vanishing geodesic distance for the Riemannian metric with geodesic equation the KdV-equation. *Ann. Global Anal. Geom.*, 41(4):461–472, 2012.

[2] M. Bauer, M. Bruveris, S. Marsland, and P. W. Michor. Constructing reparameterization invariant metrics on spaces of plane curves. *Differential Geom. Appl.*, 34:139–165, 2014.

[3] E. Celledoni, S. Eidnes, and A. Schmeding. Shape analysis on homogeneous spaces: a generalised SRVT framework. In *Computation and combinatorics in dynamics, stochastics and control*, volume 13 of *Abel Symp.*, pages 187–220. Springer, Cham, 2018.

[4] E. Celledoni, M. Eslitzbichler, and A. Schmeding. Shape analysis on Lie groups with applications in computer animation. *J. Geom. Mech.*, 8(3):273–304, 2016.



(a) From left to right: Two curves on the sphere, their original parametrizations, the reparametrization minimizing the distance in SO(3) and the reparametrization minimizing the distance in S^2, using the reductive SRVT.

(b) The interpolated curves at times $\theta = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, before reparametrization, on S^2 (blue line) and SO(3) (yellow line).

(c) The interpolated curves at times $\theta = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, after reparametrization, on S^2 (blue line) and SO(3) (yellow line).

Figure 8.2: Interpolation between two curves on S^2, with and without reparametrization, obtained by the reductive SRVT. The results are compared to the corresponding SRVT interpolation between curves on SO(3). The SO(3) curves are mapped to S^2 by multiplying with the vector $(0,1,1)^{\mathrm{T}}/\sqrt{2}$.

[5] A. Le Brigant. Computing distances and geodesics between manifold-valued curves in the srv framework, 2016.

[6] P. W. Michor and D. Mumford. Vanishing geodesic distance on spaces of submanifolds and diffeomorphisms. *Doc. Math.*, 10:217–245, 2005.

[7] P. W. Michor and D. Mumford. Riemannian geometries on spaces of plane curves. *J. Eur. Math. Soc. (JEMS)*, 8(1):1–48, 2006.

[8] A. Schmeding. Manifolds of absolutely continuous curves and the square root velocity framework, Dec. 2016.

[9] T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, Jan 2003.

[10] A. Srivastava, E. Klassen, S. Joshi, and I. Jermyn. Shape analysis of elastic curves in euclidean spaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33:1415–1428, 2011.

[11] J. Su, S. Kurtek, E. Klassen, and A. Srivastava. Statistical analysis of trajectories on Riemmannian manifolds: bird migration, hurricane tracking and video surveillance. *The Annals of Applied Statistics*, 8(2):530–552, 2014.

Shape analysis on homogeneous spaces: a generalised SRVT framework

Elena Celledoni, Sølve Eidnes and Alexander Schmeding

Published in Abel Symposia 13, pp. 187–220 (2018)

Shape analysis on homogeneous spaces: a generalised SRVT framework

Abstract. Shape analysis is ubiquitous in problems of pattern and object recognition and has developed considerably in the last decade. The use of shapes is natural in applications where one wants to compare curves independently of their parametrisation. One computationally efficient approach to shape analysis is based on the Square Root Velocity Transform (SRVT). In this paper we propose a generalised SRVT framework for shapes on homogeneous manifolds. The method opens up for a variety of possibilities based on different choices of Lie group action and giving rise to different Riemannian metrics.

9.1 Shapes on homogeneous manifolds

Shapes are unparametrised curves, evolving on a vector space, on a Lie group or on a manifold. Shape spaces and spaces of curves are infinite dimensional Riemannian manifolds, whose Riemannian metrics are the essential tool to compare and analyse shapes. By combining infinite dimensional differential geometry, analysis and computational mathematics, shape analysis provides a powerful approach to a variety of applications.

In this paper, we are concerned with the approach to shape analysis based on the Square Root Velocity Transform (SRVT), [26]. This method is effective and computationally efficient. On vector spaces, the SRVT maps parametrised curves to appropriately scaled tangent vector fields along them. The transformed curves are compared computing geodesics in the L^2 metric, and the scaling can be chosen suitably to yield reparametrisation invariance, [26], [5]. Notably, applying a (reparametrisation invariant) L^2 metric directly on the original parametrised curves is not an option as it leads to vanishing geodesic distance on parametrised curves and on the quotient shape space [4, 20]. As an alternative, higher order Sobolev type metrics were proposed [21], even though they can be computationally demanding, since computing geodesics in this infinite dimensional Riemannian setting amounts in general to solving numerically partial differential equations. These geodesics are used in practice for finding distances between curves and for interpolation between curves. The SRVT approach, on the other hand, is quite practical because it allows the use of the L^2 metric on the transformed curves: distances between curves are just L^2 distances of the transformed curves, and geodesics between curves are "*straight lines*" between the transformed curves. It is also possible to prove that this algorithmic approach corresponds (at least locally) to a particular Sobolev type metric, see [5, 9].

In the present paper we propose a generalisation of the SRVT, from vector spaces and Lie groups, [5, 26], to homogeneous manifolds. This problem has been previously considered for manifold valued curves in [18, 27], but our approach is different, the main idea is to take advantage of the Lie group acting transitively on the homogeneous manifold. The Lie group action allows us to transport derivatives of curves to our choice of base point in the homogeneous manifold. Then this information is lifted to a curve in the Lie algebra. It is natural to require that the lifted curve does not depend on the representative of the class used to pull back the curve to the base point.

The main contribution of this paper is the definition of a generalised square root velocity transform framework using transitive Lie group actions for curves on homogeneous spaces. Different choices of Lie group actions will give rise to different metrics on the infinite dimensional manifold of curves on the homogeneous space, with different properties. These different metrics, their geodesics and associated geometric tools for shape analysis can all be implemented in the computationally advantageous SRVT framework.

We extend previous results for Lie group valued curves and shapes [9], to the homogeneous manifold setting. Using ideas from the literature on differential equations on manifolds [10], we describe the main tools necessary for the definition of the SRVT and discuss the minimal requirements guaranteeing that the SRVT is well defined, section 9.2. On a general homogeneous manifold, the SRVT is obtained using a right inverse of the composition of the Lie group action with the evolution operator of the Lie group. If the homogeneous manifold is reductive, there is an explicit way to construct this right inverse (based on a canonical 1-form for the reductive space, cf. 9.3.3 - 9.3.4), see also [22]. We prove smoothness of the defined SRVT in section 9.2.1. Detailed examples on matrix Lie groups are provided in section 9.4.

A Riemannian metric on the manifold of curves on the homogeneous space is obtained by pulling back the L^2 inner product of curves on the Lie algebra through the SRVT, Theorem 9.6. To ensure that the distance function obtained on the space of parametrised curves descends to a distance function on the shape space, it is necessary to prove equivariance with respect to the group of orientation preserving diffeomorphisms (reparametrization invariance), these results are presented in section 9.2.3.

For the case of reductive homogeneous spaces, fixed the Lie group action, two different approaches are considered: one obtained pulling back the curves to the Lie algebra $\mathfrak{g}$ (Proposition 9.10) and one obtained pulling back the curves to the reductive subspace $\mathfrak{m} \subset \mathfrak{g}$ (section 9.3.6). The resulting distances are both reparametrization invariant, see Lemmata 9.8 and 9.13. For the second approach it follows similarly to what shown in [9] that the geodesic distance is globally defined by the L^2 distance, Proposition 9.12. We conjecture that also for general homogeneous manifolds, at least locally, the geodesic distance of the pullback metric is given by the L^2 distance of the curves transformed by the SRVT, see end of section 9.2.2. To illustrate the performance of the proposed approaches we compute geodesics between curves on the 2-sphere (viewed as a homogeneous space with respect to the canonical SO(3)-action), see Figure 9.1 for an example. Numerical experiments show that the two algorithms perform differently when applied to curves on the sphere (section 9.5).

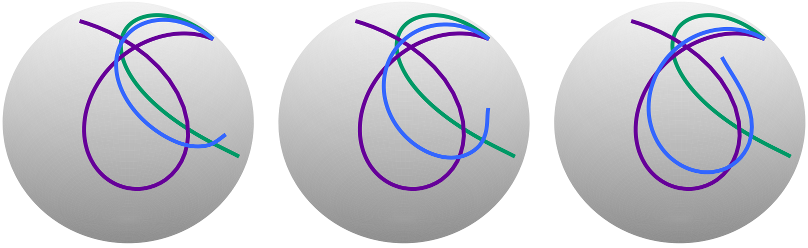


Figure 9.1: The blue curve shows the deformation of the green curve into the purple one along a geodesic $\gamma\colon [0,1] \to \operatorname{Imm}(I, S^2)$ plotted for the three times $\left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$ from left to right.

This work appeared on arXiv on the 5th of April 2017. Later a related but different work from colleagues at Florida State University was completed and posted on arXiv on the 9th of June 2017. The latter work has now appeared in [28], see also the follow-up [29]. Moreover, loc.cit. treats quotients by compact subgroups focuses on the existence of optimal reparametrisations.

9.1.1 Preliminaries and notation

Fix a Lie group G with identity element e and Lie algebra $\mathfrak{g}$.[1] Denote by $R_g\colon G \to G$ and $L_g\colon G \to G$ the right resp. left multiplication by $g \in G$. Let H be a closed Lie subgroup of G and $\mathcal{M} := G/H$ the quotient with the manifold structure turning $\pi\colon G \to G/H, g \mapsto gH$ into a submersion (see [12, Theorem G (b)]). Then $\mathcal{M}$ becomes a homogeneous space for G with respect to the (transitive) left action:

$$\Lambda\colon G \times \mathcal{M} \to \mathcal{M}, \quad (g, kH) \mapsto (gk)H.$$

For $c_0 \in \mathcal{M}$ we write $\Lambda(g, c_0) = \Lambda_{c_0}(g) = g.c_0 = \Lambda^g(c_0)$, i.e. $\Lambda_{c_0} : G \to \mathcal{M}$ (the orbit map of the orbit through c_0) and $\Lambda^g : \mathcal{M} \to \mathcal{M}$.

[1] In this paper we assume all Lie groups and Lie algebras to be finite dimensional. Note however, that many of our techniques carry over to Lie groups modelled on Hilbert spaces, [9].

9.1.2. We will consider smooth curves on $\mathcal{M}$ and describe them using the Lie group action. Namely for $c\colon [0,1] \to \mathcal{M}$ we choose a smooth lift $g\colon [0,1] \to G$ of c, i.e.:

$$c(t) = g(t).c_0, \quad c_0 \in \mathcal{M}, \quad t \in [0,1].$$

The dot denotes the action of G on G/H. In general, there are many different choices for a smooth lifts g.[2] For brevity we will in the following write $I := [0,1]$.

Later on we consider smooth functions on infinite-dimensional manifolds beyond the realm of Banach manifolds. Hence the standard definition for smooth maps (i.e. the derivative as a (continuous) map to a space of continuous operators) breaks down. We base our investigation on the so called Bastiani calculus (see [3]): A map $f\colon E \supseteq U \to F$ between Fréchet spaces is smooth if all iterated directional derivatives exist and glue together to continuous maps.[3]

9.1.3. Let M be a (possibly infinite-dimensional) manifold. By $C^\infty(I,M)$ we denote smooth functions from I to M. Recall that the topology on these spaces, the compact-open C^∞-topology, allows one to control a function and its derivatives. This topology turns $C^\infty(I,M)$ into an infinite-dimensional manifold (see e.g. [17, Section 42]).

Denote by $\mathrm{Imm}(I,M) \subseteq C^\infty(I,M)$ the set of smooth immersions (i.e. smooth curves $c\colon I \to M$ with $\dot{c}(t) \neq 0$) and recall from [17, 41.10] that $\mathrm{Imm}(I,M)$ is an open subset of $C^\infty(I,M)$.

9.1.4. We further denote by Evol the evolution operator, which is defined as

$$\begin{aligned} &\mathrm{Evol}\colon C^\infty(I,\mathfrak{g}) \to \{g \in C^\infty(I,G) : g(0) = e\} =: C^\infty_*(I,G) \\ &\mathrm{Evol}(q)(t) := g(t) \qquad \text{where} \quad \begin{cases} \frac{\mathrm{d}}{\mathrm{d}t} g &= R_{g(t)*}(q(t)), \\ g(0) &= e \end{cases} \end{aligned}$$

and $R_{g*} = T_e R_g$ is the tangent of the right translation.Recall from [13, Theorem A] that Evol is a diffeomorphism with inverse the *right logarithmic derivative*

$$\delta^r\colon C^\infty_*(I,G) \to C^\infty(I,\mathfrak{g}), \quad \delta^r g := R^{-1}_{g^*}(\dot{g}).$$

[2] Every homogeneous space G/H is a principal H-bundle, whence there are smooth horizontal lifts of smooth curves (depending on some choice of connection, cf. e.g. [23, Chapter 5.1]).

[3] In the setting of manifolds on Fréchet spaces (with which we deal here) our setting of calculus is equivalent to the so called convenient calculus (see [17]). Convenient calculus defines a map f to be smooth if it “maps smooth curves to smooth curves", i.e. $f \circ c$ is smooth for any smooth curve c. This yields a calculus on infinite-dimensional spaces where smoothness does not necessarily imply continuity (though this does not happen on Fréchet spaces), we refer to [17] for a detailed exposition. Note that both calculi can handle smooth maps on intervals $[a,b]$, see e.g. [13, 1.1] and [17, Chapter 24].

9.1.5. We fix a Riemannian metric $(\langle\cdot,\cdot\rangle_g)_{g\in G}$ on G which is right H-invariant (i.e. the maps $R_h, h \in H$ are Riemannian isometries). Since $\mathcal{M} = G/H$ is constructed using the right H-action on G, an H-right invariant metric descends to a Riemannian metric on $\mathcal{M}$. We refer to [11, Proposition 2.28] for details and will always endow the quotient with this canonical metric to relate the Riemannian geometries.

Hence H-right invariance should be seen as a minimal requirement for the metric on G. Note that a natural way to obtain (right) invariant metrics is to transport a Hilbert space inner product from the Lie algebra by (right) translation in the group. This method yields a G-right invariant metric and we will usually work with such a metric induced by $\langle\cdot,\cdot\rangle$ on $\mathfrak{g}$. Albeit it is very natural, G-invariance does not immediately add any benefits. In the following table we record properties of H, the Riemannian metric and of the canonical G-action on the quotient.

Table 9.1: Riemannian metrics and dsectionecompositions of the Lie algebra

H / $\mathfrak{h}$	**metric on** G	**special decompositions of** $\mathfrak{g}$	G**-action on** $\mathcal{M}$
compact	G-left invariant, H-biinvariant	$\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{h}^\perp$, the orthogonal complement $\mathfrak{h}^\perp$ is $\mathrm{Ad}(H)$-invariant	by isometries
compact	G-right invariant, H-biinvariant	as above	only H acts by isometries
admits reductive complement in $\mathfrak{g}$[4]	G-right invariant	$\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, $\mathfrak{m}$ is $\mathrm{Ad}(H)$-invariant $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{h}^\perp$, where in general $\mathfrak{m} \neq \mathfrak{h}^\perp$	not by isometries
	G-right invariant	$\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{h}^\perp$ but $\mathfrak{h}^\perp$ is not $\mathrm{Ad}(H)$ invariant	not by isometries

9.1.6. Let $f\colon M \to N$ be a smooth map and denote postcomposition by

$$\theta_f\colon C^\infty(I,M) \to C^\infty(I,N), \qquad c \mapsto f \circ c.$$

Note that θ_f is smooth as a map between (infinite-dimensional) manifolds.

9.1.7 (The SRVT on Lie groups)**.** For a Lie group G with Lie algebra $\mathfrak{g}$, consider an immersion $c\colon I \to G$. The square root velocity transform of c is

$$\mathcal{R}\colon \mathrm{Imm}(I,G) \to C^\infty(I,\mathfrak{g}\setminus\{0\}), \qquad \mathcal{R}(c) = \frac{\delta^r(c)}{\sqrt{\|\dot{c}\|}} = \frac{\left(R^{-1}_{c(t)}\right)_*(\dot{c})}{\sqrt{\|\dot{c}\|}}, \tag{9.1}$$

where the norm $\|\cdot\|$ is induced by a right G-invariant Riemannian metric, [9]. The SRVT consists of the composition of three maps:

[4] $\mathfrak{h} = \mathbf{L}(H)$ admits a *reductive complement* $\mathfrak{m}$, if $\mathfrak{m}$ is an $\mathrm{Ad}(H)$-invariant subspace and $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$ as vector spaces, cf. 9.3.1. Then $\mathcal{M} = G/H$ is a reductive homogeneous space.

- *differentiation* $D\colon C^\infty(I,G) \to C^\infty(I,TG), D(c) := \dot{c}$,
- *transport* $\alpha\colon C^\infty(I,TG) \to C^\infty(I,\mathfrak{g}), \quad \gamma \mapsto (R^{-1}_{\pi_{TG}\circ\gamma})_*(\gamma)$ and
- *scaling* $\mathrm{sc}\colon C^\infty(I,\mathfrak{g}\setminus\{0\}) \to C^\infty(I,\mathfrak{g}\setminus\{0\}), \quad q \mapsto \left(t \mapsto \frac{q(t)}{\sqrt{\|q(t)\|}}\right)$.

The scaling by the square root of the norm of the velocity is crucial to obtain a parametrisation invariant Riemannian metric, see [9] and Lemma 9.8.

9.2 Definition of the SRVT for homogeneous manifolds

Our aim is to construct the SRVT for curves with values in the homogeneous manifold $\mathcal{M}$. It was crucial in our investigation of the Lie group case [9] that the right-logarithmic derivative inverts the evolution operator, see 9.1.4. To mimic this behaviour we introduce a version of the evolution for homogeneous manifolds.

Definition 9.1. Fix $c_0 \in \mathcal{M}$ and denote by $C^\infty_{c_0}(I,\mathcal{M})$ all smooth curves $c\colon I \to \mathcal{M}$ with $c(0) = c_0$. Then we define

$$\rho_{c_0}\colon C^\infty(I,\mathfrak{g}) \to C^\infty_{c_0}(I,\mathcal{M}), \quad \rho_{c_0}(q) = \Lambda_{c_0}(\mathrm{Evol}(q)(t)) = \Lambda(\mathrm{Evol}(q)(t), c_0).$$

Remark 9.1. *Fix $q \in C^\infty(I,\mathfrak{g})$ and $c_0 \in \mathcal{M}$ and denote by $g(t) = \mathrm{Evol}(q)(t)$. Then*

$$\rho_{c_0}(q) := c(t) \qquad \textit{where} \qquad \begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}c(t) &= T_e\Lambda_{c(t)}(q(t)), \\ c(0) &= c_0. \end{cases}$$

Proof. In fact

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\rho_{c_0}(q)(t) &= T_{g(t)}\Lambda_{c_0}\left(\frac{\mathrm{d}}{\mathrm{d}t}g(t)\right) = T_{g(t)}\Lambda_{c_0}((R_{g(t)})_*(q(t))) \\ &= T_{g(t)}\Lambda_{c_0} \circ (R_{g(t)})_*(q(t)) = T_e(\Lambda_{c_0} \circ R_{g(t)})(q(t)) \\ &= T_e\Lambda_{\Lambda_{c_0}(g(t))}(q(t)) = T_e\Lambda_{\rho_{c_0}(q)(t)}(q(t)), \end{aligned}$$

with $T_{g(t)}\Lambda_{c_0}\colon T_{g(t)}G \to T_{\Lambda_{c_0}(g(t))}\mathcal{M} = T_{\rho_{c_0}(q)(t)}\mathcal{M}, \quad T_e\Lambda_{c(t)}\colon \mathfrak{g} \to T_{c(t)}\mathcal{M}$. □

Hence we can interpret ρ_{c_0} as a version of the evolution operator Evol for homogeneous manifolds.

Example 9.1. *Consider the two dimensional unit sphere $\mathcal{M} = S^2$ in $\mathbb{R}^3$. Consider the action of* $\mathrm{SO}(3)$ *on S^2 by matrix-vector multiplication: $\Lambda : \mathrm{SO}(3) \times S^2 \to S^2$, $\Lambda(Q,u) = Q \cdot u$. Assume $c_0 := e_1$ the first canonical vector in $\mathbb{R}^3$, then given a curve in the Lie algebra of skew-symmetric matrices $q(t) \in \mathfrak{so}(3)$, $\rho_{e_1}(q(t)) = y(t)$, where $y(t)$ satisfies $\dot{y} = q(t)y$ with $y(0) = e_1$.*

We want to construct a section of the submersion ρ_{c_0} to mimic the construction for Lie groups, see also [10, Proposition 2.2]. As we have seen in the Lie group case, the SRVT factorises into a derivation map, a map transporting the derivative to the Lie algebra and a scaling in the Lie algebra. For homogeneous spaces, we can make sense of this procedure if we can replace the transport from the Lie group case by a map which transports derivatives from the tangent bundle of the homogeneous manifold to the Lie algebra. Thus we search for a map $\alpha\colon C^\infty(I, T\mathcal{M}) \to C^\infty(I, \mathfrak{g})$ such that the following diagram commutes:

$$\begin{tikzcd}
C^\infty_{c_0}(I,\mathcal{M}) \arrow[r, "D"] \arrow[rrr, bend right, "\mathrm{id}_{C^\infty_{c_0}(I,\mathcal{M})}"] & C^\infty(I,T\mathcal{M}) \arrow[r, "\alpha"] & C^\infty(I,\mathfrak{g}) \arrow[r, "\rho_{c_0}"] & C^\infty_{c_0}(I,\mathcal{M})
\end{tikzcd}$$

Moreover, in the Lie group case we see that the mapping $\alpha \circ D$ maps the submanifold of immersions into the subset $C^\infty(I, \mathfrak{g} \setminus \{0\})$. We will require this property in general, as derivatives of immersions should vanish nowhere and this property should be preserved by the transport α. The next definition details necessary properties of α.

Definition 9.2 (Square root velocity transform)**.** Let $c_0 \in \mathcal{M}$ be fixed and define the closed submanifold[5] $\mathcal{P}_{c_0} := \{c \in \mathrm{Imm}(I,\mathcal{M}) \mid c(0) = c_0\} = \mathrm{Imm}(I,\mathcal{M}) \cap C^\infty_{c_0}(I,\mathcal{M})$ of $C^\infty(I,\mathcal{M})$. Assume there is a smooth $\alpha\colon C^\infty(I,T\mathcal{M}) \to C^\infty(I,\mathfrak{g})$, such that

$$\rho_{c_0} \circ \alpha \circ D = \mathrm{id}_{C^\infty_{c_0}(I,\mathcal{M})} \text{ and} \tag{9.2}$$

$$\alpha \circ D(\mathcal{P}_{c_0}) \subseteq C^\infty(I, \mathfrak{g} \setminus \{0\}). \tag{9.3}$$

Then we define the *square root velocity transform* on $\mathcal{M}$ at c_0, with respect to α as

$$\mathcal{R}\colon \mathcal{P}_{c_0} \to C^\infty(I, \mathfrak{g} \setminus \{0\}), \quad \mathcal{R}(c) := \frac{\alpha(\dot{c})}{\sqrt{\|\alpha(\dot{c})\|}},$$

where $\|\cdot\|$ is the norm induced by the right invariant Riemannian metric on the Lie algebra. We will see in Lemma 9.4 that $\mathcal{R}$ is smooth.

[5] As $\mathrm{Imm}(I,\mathcal{M}) \subseteq C^\infty(I,\mathcal{M})$ is open and the evaluation map $\mathrm{ev}_0\colon \mathrm{Imm}(I,\mathcal{M}) \to \mathcal{M}$ is a submersion, $\mathcal{P}_{c_0} = \mathrm{ev}_0^{-1}(c_0)$ is a closed submanifold of $\mathrm{Imm}(I,\mathcal{M})$ (cf. [12]).

The SRVT allows us to transport curves (via α) from the homogeneous manifold to curves with values in a fixed vector space (i.e. the Lie algebra $\mathfrak{g}$). *The crucial property here is that $\alpha \circ D$ is a right-inverse of ρ_{c_0}*, and we note that our construction depends strongly on the choice of the map ρ_{c_0}.

Example 9.2. *Let G be a Lie group and $H = \{e\}$ the trivial subgroup (with e the Lie group identity). Then $G = G/\{e\}$ is a homogeneous manifold and $\rho_e =$ Evol. Taking $\alpha(v) = (R_g^{-1})_*(v)$, we reproduce the definition of the SRVT on Lie groups 9.1.7. However, contrary to* Evol, *ρ_{c_0} is not invertible if the subgroup H (with $\mathcal{M} = G/H$) is non-trivial, but we might still be able to find a right inverse.*

Example 9.3. *We have $T_u S^2 := \{v \in \mathbb{R}^3 \mid v \cdot u = 0\}$ where we have denoted with "$\cdot$" the Euclidean inner product in $\mathbb{R}^3$. Then we can write*

$$v = (vu^T - uv^T)u, \qquad \forall\, v \in T_u S^2$$

and we can define the map

$$\alpha : v \in T_u S^2 \mapsto vu^T - uv^T \in \mathfrak{so}(3).$$

For c a curve evolving on S^2 with $c(0) = e_1$, we have $\rho_{e_1}(\alpha(\dot{c})) = c$, so $\alpha \circ D$ is the right inverse of ρ_{e_1}. The SRVT is then

$$\mathcal{R}(c) = \frac{\dot{c}c^T - c\dot{c}^T}{\sqrt{\|\dot{c}c^T - c\dot{c}^T\|}},$$

and $\|\cdot\|$ is the norm deduced by the usual Frobenius inner product of matrices (the scaled negative Killing form in $\mathfrak{so}(3)$ see table in example 9.5). See section 9.4 and 9.5, for further details and more examples.

The definition of α and the SRVT in Definition 9.2 depend on the initial point $c_0 \in \mathcal{M}$. In many cases our choices of α satisfy (9.2) for every $c_0 \in \mathcal{M}$, i.e. α satisfies

$$\rho(c(0), \alpha(\dot{c})) := \rho_{c(0)}(\alpha(\dot{c})) = c \quad \text{for all } c \in C^\infty(I, \mathcal{M}).$$

Further, the SRVT also depends on the choice of the left-action $\Lambda\colon G \times \mathcal{M} \to \mathcal{M}$. A different action will yield a different SRVT. For example, there are several ways to interpret a Lie group as a homogeneous manifold with respect to different group actions. One of these recovers exactly the SRVT from [9] (see Example 9.2). See [22, Section 5.1] for more information on Lie groups as homogeneous spaces, e.g. by using the Cartan-Schouten action.

Remark 9.2. *Fix $c \in C^\infty(I,\mathcal{M})$ to obtain a smooth map $\Lambda_c\colon C^\infty(I,G) \to C^\infty(I,\mathcal{M})$, $f \mapsto (t \mapsto \Lambda(f,c)(t)$ [19, Corollary 11.10 1. and Theorem 11.4]. Further we recall from [17, Theorem 42.17] that $C^\infty(I,T\mathcal{M}) \cong TC^\infty(I,\mathcal{M})$. Identifying the tangent space over the constant $e\colon I \to G$ (taking everything to the unit) we obtain*

$$T_e\Lambda_c\colon C^\infty(I,\mathfrak{g}) \to T_cC^\infty(I,\mathcal{M}), \quad q \mapsto \big(t \mapsto T_e\Lambda_{c(t)}(q(t))\big).$$

If $T_e\Lambda_c$ was invertible (which it will not be in general), we could use it to define α.

9.2.1 Smoothness of the SRVT

One of the most important properties of the square root velocity transform is that it allows us to transport curves from the manifold to curves in the Lie algebra, and this operation is smooth and invertible. The details are summarised in the following two lemmata. Following [9, Lemma 3.9], we consider the smooth scaling maps

$$\begin{aligned} \mathrm{sc}\colon C^\infty(I,\mathfrak{g}\setminus\{0\}) \to C^\infty(I,\mathfrak{g}\setminus\{0\}), \quad & q \mapsto \left(t \mapsto \frac{q(t)}{\sqrt{\|q(t)\|}}\right), \\ \mathrm{sc}^{-1}\colon C^\infty(I,\mathfrak{g}\setminus\{0\}) \to C^\infty(I,\mathfrak{g}\setminus\{0\}), \quad & q \mapsto (t \mapsto q(t)\|q(t)\|). \end{aligned} \tag{9.4}$$

Lemma 9.3. Fix $c_0 \in \mathcal{M}$, then

1. $C^\infty_{c_0}(I,\mathcal{M})$ is a closed and split submanifold[6] of $C^\infty(I,\mathcal{M})$,
2. $\rho_{c_0}\colon C^\infty(I,\mathfrak{g}) \to C^\infty_{c_0}(I,\mathcal{M})$ is a smooth surjective submersion.

Proof.

1. Note that $C^\infty_{c_0}(I,\mathcal{M})$ is the preimage of c_0 under the evaluation map
$$\mathrm{ev}_0\colon C^\infty(I,\mathcal{M}) \to \mathcal{M}, \quad c \mapsto c(0).$$
One can show, similarly to the proof of [9, Proposition 4.1] that ev_0 is a submersion.Hence, [12, Theorem C] implies that $C^\infty_{c_0}(I,\mathcal{M})$ is a closed submanifold of $C^\infty(I,\mathcal{M})$.
2. Recall that $\rho_{c_0} = \theta_{\Lambda_{c_0}} \circ \mathrm{Evol}$, with
$$\theta_{\Lambda_{c_0}}\colon C^\infty(I,G) \to C^\infty(I,\mathcal{M}), f \mapsto \Lambda_{c_0} \circ f.$$

[6] A submanifold N of a (possibly infinite-dimensional) manifold M is called *split* if it is modeled on a closed subvectorspace F of the model space E of M, such that F is complemented, i.e. $E = F \oplus G$ as topological vector spaces (see [12, Section 1]).

As $\mathcal{M}$ is a homogeneous space, $\pi\colon G \to \mathcal{M}$ is a surjective submersion. Hence [23, Chapter 5.1] implies that $\theta_\pi\colon C^\infty(I,G) \to C^\infty(I,\mathcal{M})$ is surjective. Further, the Stacey-Roberts Lemma [2, Lemma 2.4] asserts that θ_π is a submersion. Picking $g \in \pi^{-1}(c_0)$, we can also write $\theta_{\Lambda_{c_0}}(f) = \pi \circ R_g \circ f = \theta_\pi(\theta_{R_g}(f))$. Thus $\theta_{\Lambda_{c_0}} = \theta_\pi \circ \theta_{R_g}$ is a surjective submersion and

$$\theta^{-1}_{\Lambda_{c_0}}(C^\infty_{c_0}(I,\mathcal{M})) = C^\infty_*(I,G) = \{c \in C^\infty(I,G) \mid c(0) = e\}.$$

By [13, Theorem C], $\theta_{\Lambda_{c_0}}$ restricts to a smooth surjective submersion $C^\infty_*(I,G) \to C^\infty_{c_0}(I,\mathcal{M})$. Finally, since Evol: $C^\infty(I,\mathfrak{g}) \to C^\infty_*(I,G)$ is a diffeomorphism (cf. 9.1.4), $\rho_{c_0} = \theta_{\Lambda_{c_0}} \circ \mathrm{Evol}$ is a smooth surjective submersion. □

Lemma 9.4. Fix $c_0 \in \mathcal{M}$ and let α be as in Definition 9.2. Then the square root velocity transform $\mathcal{R} = \mathrm{sc} \circ \alpha \circ D$ constructed from α is a smooth immersion $\mathcal{R}\colon \mathcal{P}_{c_0} \to C^\infty(I,\mathfrak{g}\setminus\{0\})$.

Proof. The map $D\colon C^\infty(I,\mathcal{M}) \to C^\infty(I,T\mathcal{M}), c \mapsto \dot{c}$ is smooth by Lemma 9.16. Hence on $\mathcal{P}_{c_0}$, the restriction of D is smooth. As a composition of smooth maps, $\mathcal{R} = \mathrm{sc} \circ \alpha \circ D|_{\mathcal{P}_{c_0}}$ is also smooth.

Since sc: $C^\infty(I,\mathfrak{g}\setminus\{0\}) \to C^\infty(I,\mathfrak{g}\setminus\{0\})$ is a diffeomorphism, it suffices to prove that $\alpha \circ D|_{\mathcal{P}_{c_0}}$ is an immersion. As we are dealing with infinite-dimensional manifolds, it is not sufficient to prove that the derivative of $\alpha \circ D|_{\mathcal{P}_{c_0}}$ is injective (which is evident from (9.2)). Instead we have to construct immersion charts for $x \in \mathcal{P}_{c_0}$, i.e. charts in which $\alpha \circ D$ is conjugate to an inclusion of vector spaces.[7]

To construct these charts, recall from (9.2) that $f := \alpha \circ D|_{\mathcal{P}_{c_0}}$ is a right-inverse to ρ_{c_0}. In Lemma 9.3 we established that ρ_{c_0} is a surjective submersion which restricts to a submersion $\rho^{-1}_{c_0}(\mathcal{P}_{c_0}) \to \mathcal{P}_{c_0}$ by [12, Theorem C]. Fix $x \in \mathcal{P}_{c_0}$ and use the submersion charts for ρ_{c_0}. By [12, Lemma 1.2] there are open neighborhoods $x \in U_x \subseteq \mathcal{P}_{c_0}$ and $f(x) \in U_{f(x)} \subseteq \rho^{-1}_{c_0}(\mathcal{P}_{c_0})$ together with a smooth manifold N and a diffeomorphism $\theta\colon U_x \times N \to U_{f(x)}$ such that $\rho_{c_0} \circ \theta(u,n) = u$. Thus $\theta^{-1} \circ f|_{U_x} = (\mathrm{id}_{U_x}, f_2)$ for a smooth map $f_2\colon U_x \to U_{f_x}$. Hence $\theta^{-1} \circ f|_{U_x}$ induces a diffeomorphism onto the split submanifold $\Gamma(f_2) := \{(y, f_2(y)) \mid y \in U_x\} \subseteq U_x \times U_{f_x}$. Following [12, Lemma 1.13], we see that $f = \alpha \circ D|_{U_x}$ is an immersion. As x was arbitrary, the SRVT $\mathcal{R}$ is an immersion. □

Exploiting that $\mathcal{R}$ is an immersion, we transport Riemannian structures and distances from $C^\infty(I,\mathfrak{g}\setminus\{0\})$ to $\mathcal{P}_{c_0}$ by pullback. Note that the image of the SRVT for a homogeneous space is in general only an immersed submanifold

[7] See [12] for more information on immersions between infinite-dimensional manifolds.

of $C^\infty(I, \mathfrak{g} \setminus \{0\})$. For reductive homogeneous spaces, a certain SRVT will always yield a smooth embedding (see Lemma 9.11). We investigate now the Riemannian structure on $\mathcal{P}_{c_0}$.

9.2.2 The Riemannian geometry of the SRVT

As a first step, we construct a Riemannian metric using the L^2 metric on $C^\infty(I, \mathfrak{g})$.

Definition 9.5. Endow $C^\infty(I, \mathfrak{g})$ with the L^2 inner product

$$\langle f, g \rangle_{L^2} = \int_0^1 \langle f(t), g(t) \rangle \mathrm{d}t,$$

where $\langle \cdot, \cdot \rangle$ is induced by the right H-invariant Riemannian metric of G on $\mathfrak{g}$.

The L^2 inner product induces a weak Riemannian metric. The L^2-geodesics are straight lines, i.e. a curve $c(t) \in C^\infty(I, \mathfrak{g})$ is a L^2-geodesic if and only if for every t, $s \mapsto c(t)(s)$ is a straight line in the vector space $\mathfrak{g}$. In Lemma 9.4 the square root velocity transform was identified as an immersion, which we now turn into a Riemannian immersion by pulling back the L^2 metric. Arguing as in the proof of [9, Theorem 3.11] one obtains the following formula for this pullback metric.

Theorem 9.6. Let $c \in \mathcal{P}_{c_0}$ and consider $v, w \in T_c\mathcal{P}_{c_0}$, i.e. $v, w \colon I \to T\mathcal{M}$ are curves with $v(t), w(t) \in T_{c(t)}\mathcal{M}$. The pullback of the L^2 metric on $C^\infty(I, \mathfrak{g} \setminus \{0\})$ under the SRVT to the manifold of immersions $\mathcal{P}_{c_0}$ is given by:

$$\begin{aligned} G_c^{\mathcal{R}}(v, w) = \int_I \frac{1}{4} \langle D_s v, u_c \rangle \langle D_s w, u_c \rangle & \\ + \Big\langle D_s v - u_c \langle D_s v, u_c \rangle, D_s w - u_c \langle D_s w, u_c \rangle \Big\rangle \mathrm{d}s, & \end{aligned} \tag{9.5}$$

where $D_s v := T_c(\alpha \circ D)(v) / \|\alpha(\dot{c})\|$, $u_c := \alpha(\dot{c}) / \|\alpha(\dot{c})\|$ is the (transported) unit tangent vector of c, and $\mathrm{d}s = \|\alpha(\dot{c}(t))\| \mathrm{d}t$. The pullback of the L^2 norm is given by

$$G_c^{\mathcal{R}}(v, v) = \int_I \frac{1}{4} \langle D_s v, u_c \rangle^2 + \Big\| D_s v - u_c \langle D_s v, u_c \rangle \Big\|^2 \mathrm{d}s.$$

The formula for the pullback metric in Theorem 9.6 depends on α and its derivative. However, notice that we always obtain a first order Sobolev metric which measures the derivative $D_s v$ of the vector field over a curve c.

The distance on $\mathcal{P}_{c_0}$ will now be defined as the geodesic distance of the first order Sobolev metric $G^{\mathcal{R}}$, i.e. of the pullback of an L^2 metric. Thus we just need to pull the L^2 geodesic distance on $\mathcal{R}(\mathcal{P}_{c_0})$ back using the SRVT. But, in general, the geodesic distance of two curves on the submanifold $\mathcal{R}(\mathcal{P}_{c_0})$ with respect to the L^2 metric will not be the L^2 distance of the curves (see e.g. [8, Section 2]). The question is now, under which conditions is the geodesic distance at least locally given by the L^2 distance. Note first that the image of the SRVT will in general not be an open submanifold of $C^\infty(I,\mathfrak{g})$ (this was the key argument to derive the geodesic distance in [9, Theorem 3.16]). As a consequence we were unable to derive a general result describing the links between the geodesic distance by $G^{\mathcal{R}}$ on $\mathcal{P}_{c_0}$ and the SRVT algorithmic approach for homogeneous manifolds. Nonetheless, we conjecture that at least locally the geodesic distance should be given by the L^2 distance (note that $\rho_{c_0}^{-1}(\mathcal{P}_{c_0})$ is an open set, whence the geodesic distance is locally given by the L^2 distance). On the other hand, for reductive homogeneous spaces (discussed in Section 9.3), an auxiliary map can be used to obtain a geodesic distance which globally coincides with the transformed L^2 distance.

9.2.3 Equivariance of the Riemannian metric

Often in applications, one is interested in a metric on the shape space

$$\mathcal{S}_{c_0} := \mathcal{P}_{c_0} / \operatorname{Diff}^+(I) = \operatorname{Imm}_{c_0}(I,\mathcal{M}) / \operatorname{Diff}^+(I),$$

where $\operatorname{Diff}^+(I)$ is the group of orientation preserving diffeomorphisms of I acting on $\mathcal{P}_{c_0}$ from the right (cf. [6]). To assure that the distance function $d_{\mathcal{P}_{c_0}}$ descends to a distance function on the shape space, we need to require that it is invariant with respect to the group action.

Definition 9.7. Let $d\colon \mathcal{P}_{c_0} \times \mathcal{P}_{c_0} \to [0,\infty[$ be a metric. Then d is *reparametrisation invariant* if

$$d(f,h) = d(f \circ \varphi, g \circ \varphi) \quad \forall \varphi \in \operatorname{Diff}^+(I). \tag{9.6}$$

In other words d is invariant with respect to the diagonal (right) action of $\operatorname{Diff}^+(I)$ on $\mathcal{P}_{c_0} \times \mathcal{P}_{c_0}$.

Let $[f],[g] \in \mathcal{S}$ be equivalence classes and pick arbitrary representatives $f \in [f]$ and $g \in [g]$. If d is a reparametrisation invariant, we define a metric on $\mathcal{S}$ as

$$d_{\mathcal{S}}([f],[g]) := \inf_{\varphi \in \operatorname{Diff}^+(I)} d(f, g \circ \varphi). \tag{9.7}$$

Since d is reparametrisation invariant, the definition of $d_{\mathcal{S}}$ makes sense (cf. [9, Lemma 3.4]). To obtain a metric on $\mathcal{S}$, we need reparametrisation invairance of

$$d_{\mathcal{P}_{c_0}} : \mathcal{P}_{c_0} \times \mathcal{P}_{c_0} \to \mathcal{R}, \qquad d_{\mathcal{P}_{c_0}}(f,g) := \sqrt{\int_0^1 \left\| \mathcal{R}(f)(t) - \mathcal{R}(g)(t) \right\|^2 \mathrm{d}t}.$$

Lemma 9.8. Let $\mathcal{R}$ be the square root velocity transform with respect to $c_0 \in \mathcal{M}$ and $\alpha\colon C^\infty(I, T\mathcal{M}) \cong TC^\infty(I,\mathcal{M}) \to C^\infty(I,\mathfrak{g})$. Then $d_{\mathcal{P}_{c_0}}$ is reparametrisation invariant if α is a $C^\infty(I,\mathfrak{g})$-valued 1-form on $C^\infty(I,\mathcal{M})$, e.g. if $\alpha = \theta_\omega$ for a $\mathfrak{g}$-valued 1-form on $\mathcal{M}$.

Proof. Consider $\varphi \in \mathrm{Diff}^+(I)$ and $f, g \in \mathcal{P}_{c_0}$. Then a computation yields

$$\mathcal{R}(f\circ\varphi) = \frac{\alpha(\dot f\circ\varphi\cdot\dot\varphi)}{\sqrt{\left\|\alpha(\dot f\circ\varphi\cdot\dot\varphi)\right\|}} = \frac{\alpha(\dot f\circ\varphi)\cdot\dot\varphi}{\sqrt{\left\|\alpha(\dot f\circ\varphi)\cdot\dot\varphi\right\|}} = (\mathcal{R}(f)\circ\varphi)\cdot\sqrt{\dot\varphi},$$

where we have used that α is fibre-wise linear as a 1-form. Thus we can now compute

$$d_{\mathcal{P}_{c_0}}(f\circ\varphi, g\circ\varphi) = \sqrt{\int_I \left\|\mathcal{R}(f)\circ\varphi(t) - \mathcal{R}(g)\circ\varphi(t)\right\|^2 \dot\varphi(t)\mathrm{d}t} = d_{\mathcal{P}_{c_0}}(f,g). \quad \square$$

The condition on α from Lemma 9.8 is satisfied in all examples of the SRVT considered in the present paper. For example, for a reductive homogeneous case (see Section 9.3), we can always choose α as the pushforward of a $\mathfrak{g}$-valued 1-form.

9.3 SRVT for curves in reductive homogeneous spaces

A fundamental problem in our approach to shape spaces with values in homogeneous spaces is that we need to somehow lift curves from the homogeneous space to the Lie group. Ideally, this lifting process should be compatible with the Riemannian metrics on the spaces. Note that for our purposes it suffices to lift the derivatives of smooth curves to curves in the Lie algebra of the Lie group. Hence we need a suitable Lie algebra valued 1-form, which turns out to exist for reductive homogeneous spaces, cf. e.g. [16, Chapter X] (see also [22] for a recent account)

9.3.1. Recall that $\mathrm{Ad}(g) := T_e\mathrm{conj}_g$, where $\mathrm{conj}_g = L_g \circ R_{g^{-1}}$ denotes conjugation $\mathrm{conj}_g\colon G \to G$. Suppose $\mathfrak{m}$ is a subspace of $\mathfrak{g}$ such that $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$.
Let $\omega_e\colon T_{eH}\mathcal{M} \to \mathfrak{m}$ be the inverse of $T_e\pi|_{\mathfrak{m}}\colon \mathfrak{g} \supseteq \mathfrak{m} \to T_{eH}\mathcal{M}$. Identify $\mathfrak{g} = T_eG$ and observe that $T_e\pi\colon \mathfrak{g} \to T_{eH}\mathcal{M}$ induces an isomorphism $T_e\pi|_{\mathfrak{m}}\colon \mathfrak{m} \to T_{eH}\mathcal{M}$.

By definition $\pi \circ R_h = \pi$ holds for all $h \in H$. Now the group actions of G on itself by left and right multiplication commute and we observe that, for all $g \in G$,

$$\pi \circ L_g = \Lambda^g \circ \pi \quad \text{and} \quad T_e\pi \circ \mathrm{Ad}(h) = T\Lambda^h \circ T_e\pi \text{ for } h \in \mathfrak{h}. \tag{9.8}$$

9.3.2. We will from now on assume that $\mathcal{M}$ is a reductive homogeneous manifold. This means that the subalgebra $\mathfrak{h}$ admits a *reductive complement*, i.e. a vector subspace $\mathfrak{m} \subseteq \mathfrak{g}$ such that

$$\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m} \text{ and } \mathrm{Ad}(h).\mathfrak{m} \subseteq \mathfrak{m} \text{ for all } h \in H.$$

If it exists, a reductive complement will in general not be unique. However, we choose and fix a reductive complement $\mathfrak{m}$ for $\mathfrak{h}$.

9.3.3. As a reductive complement, $\mathfrak{m}$ is closed with respect to the adjoint action of H. Hence one deduces (cf. [22, Lemma 4.6] for a proof) that ω_e is H-invariant with respect to the adjoint action, i.e.

$$\omega_e(T\Lambda^h(v)) = \mathrm{Ad}(h).\omega_e(v) \quad \text{for all } v \in T_{eH}\mathcal{M} \text{ and } h \in H.$$

Thus the following map is well-defined:

$$\omega\colon T\mathcal{M} \to \mathfrak{g}, \quad v \mapsto \mathrm{Ad}(g).\omega_e(T\Lambda^{g^{-1}}(v)) \quad \text{for all } v \in T_{gH}\mathcal{M}.$$

From the definition it is clear that ω is a smooth $\mathfrak{g}$-valued 1-form on $\mathcal{M}$. Moreover, ω is even G-equivariant with respect to the canonical and adjoint action:

$$\omega(T\Lambda^k(v)) = \mathrm{Ad}(k).\omega(v) \quad \text{for all } v \in T\mathcal{M} \text{ and } k \in G. \tag{9.9}$$

Note that ω depends by construction on our choice of reductive complement $\mathfrak{m}$. However, we will suppress this dependence in the notation. As noted in [22, Section 4.2], the 1-forms ω correspond bijectively to reductive structures on G/H.[8]

9.3.4. Let ω be the 1-form constructed in 9.3.3. Then we define the map

$$\theta_\omega\colon C^\infty(I, T\mathcal{M}) \to C^\infty(I, \mathfrak{g}), \quad f \mapsto \omega \circ f.$$

Note first that θ_ω is smooth by [17, Theorem 42.13]. We will prove that θ_ω indeed satisfies (9.2) and (9.3), whence $\alpha = \theta_\omega$ yields an SRVT as in 9.2.

To motivate the computations, let us investigate an important special case.

Example 9.4. *Similarly to example 9.2, let G be a Banach Lie group and $H = \{e\}$ the trivial subgroup. Then $G = G/\{e\}$ can be viewed as a reductive homogeneous manifold with* $\mathfrak{m} = \mathfrak{g}$, $\pi = \mathrm{id}_G$ *and* $\omega_e = \mathrm{id}_{\mathfrak{g}}$. *From the definition of ω we obtain* $\omega(v) = \mathrm{Ad}(g).(L^{g^{-1}})_*(v) = (R_g^{-1})_*(v) = \kappa^r(v)$, *where κ^r denotes the right Maurer-Cartan form, [17, Section 38] or [22, Section 5.1]. In particular, for $c\colon I \to G$ we have $\theta(c) = \kappa^r(\dot{c}) = \delta^r(c)$ (right logarithmic derivative). As we have* $\mathrm{Evol} \circ \delta^r(c) = c$ *for a curve starting at e.*

The SRVT for reductive spaces coincides thus with the SRVT for Lie group valued shape spaces as outlined in 9.1.7.

[8] Note that there might be different reductive structures on a homogeneous manifold. We refer to [22, Section 5.1] for examples and further references.

Albeit Example 9.4 is quite trivial as a homogeneous space, it highlights a general principle of the construction for reductive homogeneous spaces.

Remark 9.3. *We here provide an alternative interpretation for $\theta_\omega \circ D$: A smooth curve $c\colon I \to \mathcal{M}$ admits a smooth horizontal lift $\tilde{c}\colon I \to G$ depending on a choice of connection for the principal bundle $G \to \mathcal{M}$ [23, Chapter 5.1]. For a reductive homogeneous manifold we construct a horizontal lift $\tilde{c}$ using the canonical invariant connection (depending on the reductive complement, see [16, X.2]). Now we take the (right) Darboux derivative (aka right logarithmic derivative) of $\tilde{c}\colon I \to G$ (see [25, 3.§5]). Then unraveling the definitions similar to Examples 9.2 and 9.4, one can show that $\delta^r(\tilde{c}) = \theta_\omega \circ D(c)$ holds for the 1-form θ_ω as in 9.3.4. Thus for a reductive homogeneous space the proposed SRVT can be viewed (up to scaling) as the Darboux derivative of a horizontal lift of a curve in $\mathcal{M}$. Note that this interpretation justifies again to view ρ_{c_0} as a generalised version of the evolution operator* Evol *(which inverts the right logarithmic derivative, see Remark 9.1).*

A rich source for reductive homogeneous spaces are quotients of semisimple Lie groups. We recall now some of the main examples.

Example 9.5. *Let G be a semisimple Lie group and H a Lie subgroup of G which is also semisimple. Then the homogeneous space $\mathcal{M} = G/H$ is reductive. A reductive complement of $\mathfrak{h}$ in $\mathfrak{g}$ is the orthogonal complement $\mathfrak{h}^\perp$ with respect to the Cartan-Killing form on $\mathfrak{g}$ (recall that the Killing form of a semisimple Lie algebra is non-degenerate by Cartan's criterion [15, I.§7 Theorem 1.45]). For example, this occurs for $G = \mathrm{SL}(n)$ and $H = \mathrm{SL}(n-p)$ or $G = \mathrm{SO}(n)$ and $H = \mathrm{SO}(n-p)$ (where $1 \leq p < n$), since by [15, I.§8 and I.§18] the following properties hold:*

Lie group G	**compact?**	**semisimple?**	**Killing form** $B(X,Y)$ **on** $\mathfrak{g}$
$\mathrm{SO}(n)$	*yes*	*yes (for $n \geq 3$)*	$(n-2)Tr(XY)$
$\mathrm{SL}(n)$	*no*	*yes*	$2nTr(XY)$
$\mathrm{GL}(n)$	*no*	*no*	$2nTr(XY) - 2Tr(X)Tr(Y)$

Here Tr denotes the trace of a matrix. All main examples in this paper are reductive.

Proposition 9.9. Let $\mathcal{M} = G/H$ be a reductive homogeneous space, $c_0 \in \mathcal{M}$, ω and θ_ω as in 9.3.4. Consider $D\colon C^\infty_{c_0}(I,\mathcal{M}) \to C^\infty(I,T\mathcal{M}), c \mapsto \dot{c}$. Then

$$\rho_{c_0} \circ \theta_\omega \circ D = \mathrm{id}_{C^\infty_{c_0}(I,\mathcal{M})}.$$

Proof. As a shorthand write $\theta := \theta_\omega \circ D$. We establish in Lemma 9.17 the identity

$$\mathrm{id}_{C^\infty_{eH}(I,\mathcal{M})} = \rho_{eH} \circ \theta = \Lambda_{eH} \circ \mathrm{Evol} \circ \theta = \pi \circ \mathrm{Evol} \circ \theta. \tag{9.10}$$

Let now $c \in C^\infty_{c_0}(I,\mathcal{M})$ with $c_0 = g_0 H$. Then we obtain $\Lambda^{g_0^{-1}} \circ c \in C^\infty_{eH}(I,\mathcal{M})$ and

$$\begin{aligned}\rho_{c_0} \circ \theta(c) &= (\Lambda_{c_0} \circ \mathrm{Evol}) \circ \theta_\omega(\dot{c}) = \Lambda_{c_0} \circ \mathrm{Evol} \circ \omega(T\Lambda^{g_0} T\Lambda^{g_0^{-1}} \dot{c}) \\ &\overset{(9.9)}{=} \Lambda_{c_0} \circ \mathrm{Evol}(\mathrm{Ad}(g_0).\omega(T\Lambda^{g_0^{-1}} \dot{c})) = \Lambda_{c_0} \circ \mathrm{Evol}(\mathrm{Ad}(g_0).\theta(\Lambda^{g_0^{-1}} \circ c)).\end{aligned}$$

Recall from [13, 1.16] that for a Lie group morphism φ one has the identity $\mathrm{Evol} \circ \mathbf{L}(\varphi) = \varphi \circ \mathrm{Evol}$. By definition, $\mathrm{Ad}(g) = \mathbf{L}(\mathrm{conj}_g) := T_e \mathrm{conj}_g$, where $\mathrm{conj}_g = L_g \circ R_{g^{-1}}$ denotes the conjugation morphism. Insert this into the above equation:

$$\begin{aligned}\rho_{c_0} \circ \theta(c) = \Lambda_{c_0} \circ \mathrm{Evol} \circ \theta(c) &= \Lambda_{c_0} \circ L_{g_0} \circ R_{g_0^{-1}} \circ \mathrm{Evol}(\theta(\Lambda^{g_0^{-1}} \circ c)) \\ &= \pi \circ L_{g_0} \mathrm{Evol}(\theta(\Lambda^{g_0^{-1}} \circ c)) = \Lambda^{g_0} \circ \pi \circ \mathrm{Evol}(\theta(\Lambda^{g_0^{-1}} \circ c)) \\ &\overset{(9.10)}{=} \Lambda^{g_0} \circ \Lambda^{g_0^{-1}} \circ c = c.\end{aligned}$$

In passing to the second line we used that left and right multiplication maps commute and that $\Lambda_{c_0}(R_{g_0^{-1}}(k)) = \Lambda_{c_0}(kg_0^{-1}) = kg_0^{-1}c_0 = kg_0^{-1}g_0 H = \pi(k)$. □

Proposition 9.10. Let $\mathcal{M} = G/H$ be a reductive homogeneous space, $c_0 \in \mathcal{M}$, ω and θ_ω as in 9.3.4. Then θ_ω satisfies (9.2) and (9.3), whence for a reductive homogeneous space we can define the SRVT as

$$\mathcal{R}(c) := \frac{\theta_\omega(\dot{c})}{\sqrt{\|\theta_\omega(\dot{c})\|}} \qquad \text{for } c \in \mathrm{Imm}(I,\mathcal{M})$$

Proof. In Proposition 9.9 we have already established (9.2). To see that (9.3) also holds for θ_ω, observe first that for $v \in T_{gH}\mathcal{M}$, we have $\omega(v) = \mathrm{Ad}(g).\omega_e(T\Lambda^{g^{-1}}(v))$. Since $\omega_e \circ T\Lambda^{g^{-1}} \colon T_{gH}\mathcal{M} \to \mathfrak{m}$ and $\mathrm{Ad}(g) \colon \mathfrak{g} \to \mathfrak{g}$ are linear isomorphisms, we see that $\omega(v) = 0$ if and only if $v = 0_{gH}$. As θ_ω is post-composition by ω, θ_ω satisfies (9.3). □

9.3.5 Riemannian geometry and the reductive SRVT

In the reductive space case, it is easier to describe the image of the square root velocity transform. It turns out that the image is a split submanifold with a global chart. Using this chart, we can also obtain information on the geodesic distance.

The idea is to transform the image of the SRVT such that it becomes $C^\infty(I, \mathfrak{m} \setminus \{0\})$, where $\mathfrak{m}$ is again the reductive complement. Pick $g_0 \in \pi^{-1}(c_0)$ and use the adjoint action of G and the evolution $\mathrm{Evol} \colon C^\infty(I,\mathfrak{g}) \to C^\infty(I,G)$ to define

$$\Psi_{g_0}(q) := -\mathrm{Ad}(g_0 \mathrm{Evol}(q)^{-1}).q \qquad \text{for } q \in C^\infty(I,\mathfrak{g})$$

where the dot denotes pointwise application of the linear map $\mathrm{Ad}(\mathrm{Evol}(q)^{-1})$. Then Ψ_{g_0} is a diffeomorphism with inverse $\Psi_{g_0^{-1}}$ (see Lemma 9.19). We will now see that $\Psi_{g_0^{-1}}$ maps the image of the SRVT to $C^\infty(I, \mathfrak{m} \setminus \{0\})$.

Lemma 9.11. Choose $c_0 \in \mathcal{M}$ in the reductive homogeneous space $\mathcal{M}$, and let ω and θ_ω, D be as in Proposition 9.9. Then $\operatorname{Im} \theta_\omega \circ D$ is a split submanifold of $C^\infty(I, \mathfrak{g} \setminus \{0\})$ modelled on $C^\infty(I, \mathfrak{m})$ and $\theta_\omega \circ D$ is a smooth embedding. In particular, $\mathcal{R}(\mathcal{P}_{c_0}) = \Psi_{g_0}(C^\infty(I, \mathfrak{m} \setminus \{0\}))$ is a split submanifold of $C^\infty(I, \mathfrak{g} \setminus \{0\})$ and $\mathcal{R}$ is a smooth embedding.

Proof. As $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, we have $C^\infty(I, \mathfrak{g}) = C^\infty(I, \mathfrak{h} \oplus \mathfrak{m}) \cong C^\infty(I, \mathfrak{h}) \oplus C^\infty(I, \mathfrak{m})$. Thus $C^\infty(I, \mathfrak{m} \setminus \{0\})$ is a closed and split submanifold of $C^\infty(I, \mathfrak{g} \setminus \{0\})$. Fix $g_0 \in G$ with $\pi(g_0) = c_0$ and note that Ψ_{g_0} restricts to a diffeomorphism $C^\infty(I, \mathfrak{g} \setminus \{0\}) \to C^\infty(I, \mathfrak{g} \setminus \{0\})$ by Lemma 9.19. Now as $\Psi_{g_0}(C^\infty(I, \mathfrak{m} \setminus \{0\})) = \operatorname{Im} \theta_\omega \circ D$ (cf. Lemma 9.20), the image $\operatorname{Im} \theta_\omega \circ D$ is a closed and split submanifold of $C^\infty(I, \mathfrak{g} \setminus \{0\})$. Further, we deduce from Lemma 9.20 that $\rho_{c_0}|_{\operatorname{Im} \theta_\omega \circ D}$ is smooth with $\theta_\omega \circ D \circ \rho_{c_0}|_{\operatorname{Im} \theta_\omega \circ D} = \mathrm{id}_{\operatorname{Im} \theta_\omega \circ D}$. As also $\rho_{c_0} \circ \theta_\omega = \mathrm{id}_{\mathrm{Imm}_{c_0}(I, \mathcal{M})}$, we see that θ_ω is a diffeomorphism onto its image. Thus $\theta_\omega \circ D$ is indeed a smooth embedding.

Since the scaling maps are diffeomorphisms $C^\infty(I, \mathfrak{g} \setminus \{0\}) \to C^\infty(I, \mathfrak{g} \setminus \{0\})$, the assertions on the image of $\mathcal{R}$ and on $\mathcal{R}$ follow directly from the assertions on θ_ω. □

9.3.6 (Reductive SRVT)**.** Let $\mathcal{M}$ be a reductive homogeneous space with reductive complement $\mathfrak{m}$ and $\theta_\omega \colon C^\infty(I, T\mathcal{M}) \to C^\infty(I, \mathfrak{g})$, $f \mapsto \omega \circ f$ be constructed with respect to the 1-form ω from 9.3.3. Then $\Psi_{g_0^{-1}} \circ \theta_\omega(\mathcal{P}_{c_0}) = C^\infty(I, \mathfrak{m} \setminus \{0\})$ (see Appendix 9.7.1). Now one constructs a version of the SRVT for reductive spaces via

$$\mathcal{R}_\mathfrak{m} \colon \mathcal{P}_{c_0} \to C^\infty(I, \mathfrak{m} \setminus \{0\}), \quad f \mapsto \frac{\Psi_{g_0^{-1}} \circ \theta_\omega(\dot{f})}{\sqrt{\|\Psi_{g_0^{-1}} \circ \theta_\omega(\dot{f})\|}}.$$

We call this map *reductive SRVT*, to distinguish it from the usual SRVT. Contrary to the SRVT, the reductive SRVT will go into the reductive complement, but it will not be a section of ρ_{c_0}. Instead it is a section of $\rho_{c_0} \circ \Psi_{g_0}$. Finally, we note that by construction (cf. Lemma 9.19) the image of the reductive SRVT is $C^\infty(I, \mathfrak{m} \setminus \{0\})$.

Arguing as in Theorem 9.6, we also obtain a first order Sobolev metric by pullback with the reductive SRVT. In general this Riemannian metric will not coincide with the pullback metric obtained from the SRVT. The advantage of the reductive SRVT is that we have full control over its image, which happens to be an open subset (of a subspace of $C^\infty(I,\mathfrak{g})$). Since $C^\infty(I,\mathfrak{g})$ with respect to the L^2 inner product is a flat space (in the sense of Riemannian geometry), it follows that at least locally the geodesic distance on the image of the SRVT is given by the distance

$$d_{\mathcal{P}_{c_0},\mathfrak{m}}(f,g) := d_{L^2}(\mathcal{R}_\mathfrak{m}(f),\mathcal{R}_\mathfrak{m}(g)).$$

However, we argue as in [9, Theorem 3.16] to obtain the following result.

Proposition 9.12. If dim $\mathfrak{h}+2 <$ dim $\mathfrak{g}$, then the geodesic distance of $(\mathcal{R}(\mathcal{P}_{c_0}),\langle\cdot,\cdot\rangle_{L^2})$ coincides with the L^2 distance. In this case the geodesic distance on $\mathcal{P}_{c_0}$ induced by the pullback metric (9.5) (with respect to the reductive SRVT) is given by $d_{\mathcal{P}_{c_0},\mathfrak{m}}(f,g) = \sqrt{\int_I \left\|\mathcal{R}_\mathfrak{m}(f)(t) - \mathcal{R}_\mathfrak{m}(g)(t)\right\|^2 \mathrm{d}t}$.

Note that the modification by the reductive SRVT is highly non-linear, e.g. in the Lie group case, Example 9.4, we obtain:

Example 9.6. *Let G be a Lie group, $c \in^\infty (I,G)$ and $\delta^l(c) = c^{-1}\dot{c}$. Then*

$$\Psi(\delta^r(c)) = -\mathrm{Ad}(\mathrm{Evol}(\delta^r(c))^{-1}).\delta^r(c) = -\mathrm{Ad}(c^{-1}).\dot{c}c^{-1} = -\delta^l(c).$$

Recall from [17, 38.4] that $\mathrm{Evol}(-\delta^l(c))(t) = (c(t))^{-1}$. *In the Lie group case, the reductive SRVT modifies the formulae to compute distances and interpolations between the pointwise inverses of curves instead of the curves themselves. In particular, this shows that the reductive SRVT will not be a section of ρ_{c_0}.*

In particular, we have to prove a version of Lemma 9.8 for the reductive SRVT.

Lemma 9.13. For a reductive space, $d_{\mathcal{P}_{c_0},\mathfrak{m}}$ is reparametrisation invariant.

Proof. For $\mathcal{R}_\mathfrak{m}$ we use $\Psi_{g_0^{-1}} \circ \theta_\omega$ instead of $\alpha = \theta_\omega$. Consider $f \in \mathcal{P}_{c_0}$ and $\varphi \in \mathrm{Diff}^+(I)$ to compute as in Lemma 9.8: $\Psi_{g_0}^{-1}(\theta_\omega(f\circ\varphi)) = \Psi_{g_0}^{-1}(\dot{\varphi}\cdot\theta_\omega(f)\circ\varphi)$. Now

$$\mathrm{Evol}(q)\circ\varphi = \mathrm{Evol}(\dot{\varphi}\cdot q\circ\varphi)\underbrace{\mathrm{Evol}(q)(\varphi(0))}_{=e \text{ since } \varphi(0)=0} = \mathrm{Evol}(\dot{\varphi}\cdot q\circ\varphi)$$

follows from [17, p. 411]. Linearity of the adjoint action yields $\Psi_{g_0}^{-1}(\theta_\omega(f\circ\varphi)) = (\Psi_{g_0}^{-1}(\theta_\omega(f))\circ\varphi)\cdot\dot{\varphi}$. Inserting this in (9.9) yields reparametrisation invariance. □

9.4 The SRVT on matrix Lie groups

In order to illustrate our definition of the SRVT in different instances of homogeneous manifolds, we consider in what follows two examples of quotients of finite dimensional matrix Lie groups (for $n \geq 3$ and $p < n$):

1. $\mathrm{SO}(n)/(\mathrm{SO}(n-p) \times \mathrm{SO}(p))$ (see 9.4.3).
2. $\mathrm{SO}(n)/\,\mathrm{SO}(n-p)$ (see 9.4.2).

Note that in both cases the quotients are reductive homogeneous spaces. To prepare our investigation, we will now collect some information on relevant tangent spaces for the matrix Lie groups. These examples are relevant in applications [1].

9.4.1 Tangent space of G/H and tangent map of $G \to G/H$

For G and H finite dimensional (matrix) Lie groups, we here describe the tangent space of G/H at a prescribed point c_0 and the tangent mapping of the canonical projection $\pi : G \to G/H$. We have seen that any curve $c(t)$ on G/H, $c(0) = c_0$, can be expressed non-uniquely by means of a curve on the Lie group $c(t) = \pi(g(t))$. For matrix Lie groups, the elements of G/H are equivalence classes of matrices. Let the elements of G, $g \in G$, be $n \times n$ matrices, then the group multiplication coincides with matrix multiplication. We identify elements of $H \subset G$ with matrices

$$h = \begin{bmatrix} I & 0 \\ 0 & \Gamma \end{bmatrix}, \tag{9.11}$$

where Γ is an $(n-p) \times (n-p)$ matrix and I is the $p \times p$ identity.

We obtain $T_{g_0}\pi : T_{g_0}G \to T_{\pi(g_0)}G/H$, $v \mapsto w$, by differentiating $c(t) = \pi(g(t))$. Assuming $g(0) = g_0$, $\pi(g_0) = c_0$, $\dot{g}(0) = v \in T_{g_0}G$, we have

$$w := T_{\pi(g_0)}(v) = \left.\frac{d}{dt}\right|_{t=0} \pi(g(t)) = \left\{ \left.\frac{d}{dt}\right|_{t=0} \tilde{g}(t) \,|\, \tilde{g}(t) = g(t)\,h(t), \quad h(t) \in H \right\}.$$

Assuming $\dot{g}(t) = A(t)g(t)$, where $A(t) \in \mathfrak{g}$, $v = A_0 g_0 = g_0 \mathrm{Ad}_{g_0^{-1}}(A_0)$, and assuming also that $\frac{d}{dt}h(t) = B(t)h(t)$, $B(t) \in \mathfrak{h}$, $B(0) = B_0$, in analogy to (9.32), we get

$$\begin{aligned} \frac{d}{dt}\tilde{g}(t) &= \left(A(t) + \mathrm{Ad}_{g(t)}(B(t))\right) g(t)h(t) \\ &= g(t)\left(\mathrm{Ad}_{g(t)^{-1}}(A(t)) + B(t)\right) h(t), \end{aligned} \tag{9.12}$$

so we obtain

$$w := T_{\pi(g_0)}(v) = \left\{ \tilde{w} = \frac{d}{dt}\bigg|_{t=0} \tilde{g}(t) \mid \tilde{w} = (A_0 + \mathrm{Ad}_{g_0}(B_0))\, g_0\, h, \quad h \in H, B_0 \in \mathfrak{h} \right\}$$

$$= \left\{ \tilde{w} = \frac{d}{dt}\bigg|_{t=0} \tilde{g}(t) \mid \tilde{w} = g_0\, (\mathrm{Ad}_{g_0^{-1}}(A_0) + B_0)\, h, \quad h \in H, B_0 \in \mathfrak{h} \right\},$$

which gives a description of the tangent vector $w \in T_{c_0}G/H$ as well as the characterisation of $T\pi$ for matrix Lie groups. Suppose that we fix a complementary subspace $\mathfrak{m}$ of $\mathfrak{h}$, $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, then there is a unique isotropy element $B_0 \in \mathfrak{h}$ such that $\mathrm{Ad}_{g_0^{-1}}(A_0) + B_0 \in \mathfrak{m}$.

Repeating this procedure for each value of t along a curve $c(t)$, we can assume $c(t) = \pi(g(t))$ and $w(t) \in T_{\pi(g(t))}G/H$, $w(t) = (A(t) + \mathrm{Ad}_{g(t)}(B(t)))c(t)$ with $A(t) \in \mathfrak{g}$, $B(t) \in \mathfrak{h}$, such that $\mathrm{Ad}_{g(t)^{-1}}(A(t)) + B(t) \in \mathfrak{m}$, then we can define

$$\alpha : T_{\pi(g(t))}G/H \to \mathrm{Ad}_{g(t)}(\mathfrak{m}), \qquad \alpha(w(t)) = A(t) + \mathrm{Ad}_{g(t)}(B(t)).$$

This map corresponds to the map θ_ω of 9.3.4 with ω as described in 9.3.3. If $\mathfrak{m}$ is reductive, this map is well defined (independently of the choice of representative $g(t)$ of $c(t) = \pi(g(t))$). We refer to Table 9.1 for different, possible choices of $\mathfrak{m}$ and their implications. In the following examples $\mathfrak{m}$ is reductive and H is compact.

9.4.2 SRVT on the Stiefel manifold: $\mathrm{SO}(n)/\mathrm{SO}(n-p)$.

In this section we consider the case when $G = \mathrm{SO}(n)$ and $H = \mathrm{SO}(n-p) \subset \mathrm{SO}(n)$, where the elements of $\mathrm{SO}(n-p)$ are of the type (9.11) with Γ a $(n-p) \times (n-p)$ orthogonal matrix with determinant equal to 1. We consider the canonical left action of $\mathrm{SO}(n)$ on the quotient $\mathrm{SO}(n)/\mathrm{SO}(n-p)$. This homogeneous manifold can be identified with the Stiefel manifold, $\mathcal{M} = \mathbb{V}_p(\mathbb{R}^n)$, i.e. the set of p-orthonormal frames in $\mathbb{R}^n$ (real matrices $n \times p$ with orthonormal columns). We will in the following denote by $[U, U^\perp]$ the elements of $\mathrm{SO}(n)$ where we have collected in U the first p orthonormal columns and in $U^\perp$ the last $n-p$. Multiplication from the right by an arbitrary element in the isotropy subgroup $\mathrm{SO}(n-p)$ gives $[U, U^\perp\Gamma]$, leaving the first p columns unchanged and orthonormal to the last $n-p$, for all choices of Γ. Here U alone represents the whole coset of $[U, U^\perp]$. When thought of as a map from $\mathrm{SO}(n)$ to $\mathrm{SO}(n)/\mathrm{SO}(n-p)$, the projection π: $\mathrm{SO}(n) \to \mathrm{SO}(n)/\mathrm{SO}(n-p)$ is

$$\pi([U, U^\perp]) = \{\tilde{g} \in \mathrm{SO}(n) \mid \tilde{g} = [U, U^\perp\Gamma], \quad \forall\, \Gamma \in \mathrm{SO}(n-p)\}.$$

Otherwise, when thought of as a map from $\pi : \mathrm{SO}(n) \to \mathbb{V}_p(\mathbb{R}^n)$, the canonical projection conveniently becomes $\pi([U, U^\perp]) = [U, U^\perp]\, I_p = U$, where I_p is the

$n \times p$ matrix whose columns are the first p columns of the $n \times n$ identity matrix. The equivalence class of the group identity element $\pi(e)$ is identified with the $n \times p$ matrix I_p. Similarly the tangent mapping of the projection π,

$$T\pi : T\mathrm{SO}(n) \to T\mathrm{SO}(n)/\mathrm{SO}(n-p),$$

$v \in T_g\mathrm{SO}(n) \mapsto w \in T_{\pi(g)}\mathrm{SO}(n)/\mathrm{SO}(n-p)$, with $g = [U, U^{\perp}]$, $v = A[U, U^{\perp}] \in T_{[U,U^{\perp}]}\mathrm{SO}(n)$ and $A \in \mathfrak{so}(n)$), can be realised as

$$\begin{aligned} &T_{[U,U^{\perp}]}\pi(A[U,U^{\perp}]) \\ &\quad = \left\{ \tilde{w} \in T_{[U,U^{\perp}\Gamma]}\,\mathrm{SO}(n) \,\middle|\, \begin{array}{c} \tilde{w} = [U,U^{\perp}](\mathrm{Ad}_{[U,U^{\perp}]^T}(A) + B)\Gamma, \\ \forall \Gamma \in \mathrm{SO}(n-p),\, B \in \mathfrak{so}(n-p) \end{array} \right\}, \end{aligned} \tag{9.13}$$

while $T\pi : T\mathrm{SO}(n) \to T\mathbb{V}_p(\mathbb{R}^n)$ by multiplication from the right by I_p, and

$$T_{[U,U^{\perp}]}\pi(A[U,U^{\perp}]) = A[U,U^{\perp}]\,I_p = AU \in T_U\mathbb{V}_p(\mathbb{R}^n). \tag{9.14}$$

Alternatively, a characterisation of tangent vectors can be obtained by differentiation of curves on $\mathbb{V}_p(\mathbb{R}^n)$. We have then that

$$T_Q\mathbb{V}_p(\mathbb{R}^n) = \{V \quad n \times p \;\text{ matrix} \mid Q^{\mathrm{T}}V \quad p \times p \quad \text{skew-symmetric}\}.$$

Proposition 9.14. [10] Any tangent vector V at $Q \in \mathbb{V}_p(\mathbb{R}^n)$ can be written as

$$V = (FQ^{\mathrm{T}} - QF^{\mathrm{T}})\,Q, \tag{9.15}$$

$$F := V - Q\frac{Q^{\mathrm{T}}V}{2} \in T_Q\mathcal{M}. \tag{9.16}$$

And notice that replacing F with $F := V - Q(\frac{Q^{\mathrm{T}}V}{2} + S)$, where S is an arbitrary $p \times p$ symmetric matrix, does not affect (9.15).

We proceed by using the representation (9.15) of $T_Q\mathbb{V}_p(\mathbb{R}^n)$ and the framework described in Definition 9.2 for defining an SRVT on the Stiefel manifold. Consider

$$f_Q : T_Q\mathcal{M} \to T_Q\mathcal{M}, \qquad f_Q(V) = V - Q\frac{Q^{\mathrm{T}}V}{2}, \tag{9.17}$$

$$a_Q : T_Q\mathcal{M} \to \mathfrak{m}_Q \subset \mathfrak{g}, \qquad a_Q(V) = f_Q(V)Q^{\mathrm{T}} - Qf_Q(V)^{\mathrm{T}}. \tag{9.18}$$

The SRVT of a curve $Y(t)$ on the Stiefel manifold is a curve on $\mathfrak{so}(n)$ defined by

$$\mathcal{R}(Y) := \frac{a_Y(\dot{Y})}{\sqrt{\|a_Y(\dot{Y})\|}} = \frac{f_Y(\dot{Y})Y^{\mathrm{T}} - Yf_Y(\dot{Y})^{\mathrm{T}}}{\sqrt{\|f_Y(\dot{Y})Y^{\mathrm{T}} - Yf_Y(\dot{Y})^{\mathrm{T}}\|}}. \tag{9.19}$$

As the Stiefel manifold is a reductive homogeneous space, we can define a reductive SRVT in this case. Denoting with $[Q, Q^{\perp}]$ a representative in $SO(n)$ of the equivalence class identified by Q on $\mathbb{V}_p(\mathbb{R}^n)$, we observe that

$$V = \mathrm{Ad}_{[QQ^{\perp}]}(G)I_p \quad \text{with} \quad G := [QQ^{\perp}]^T F I_p^T - I_p F^T [QQ^{\perp}].$$

Assuming the right invariant metric on $SO(n)$ is the negative Killing form, then we observe that G belongs to the orthogonal complement of the subalgebra $\mathfrak{so}(n-p)$ in $\mathfrak{so}(n)$ with respect to this inner product. As stated in Table 9.1, this orthogonal complement is the reductive complement, i.e. $\mathfrak{m} = \mathfrak{so}(n-p)^{\perp}$, and $\mathrm{Ad}_{SO(n-p)}(\mathfrak{so}(n-p)^{\perp}) \subset \mathfrak{so}(n-p)^{\perp}$. The elements of such an orthogonal complement $\mathfrak{so}(n-p)^{\perp}$ are matrices $W \in \mathfrak{so}(n)$ of the form

$$W = \begin{bmatrix} \Omega & \Sigma^T \\ -\Sigma & 0 \end{bmatrix}, \tag{9.20}$$

with $\Omega \in \mathfrak{so}(p)$ and Σ an arbitrary $(n-p) \times p$ matrix. Consider the maps

$$\tilde{f}_Q : T_Q\mathcal{M} \to T_{I_p}\mathcal{M}, \qquad \tilde{f}_Q(V) = [QQ^{\perp}]^T V - I_p \frac{Q^{\mathrm{T}} V}{2}, \tag{9.21}$$

$$\tilde{a}_Q : T_Q\mathcal{M} \to \mathfrak{m} \subset \mathfrak{g}, \qquad \tilde{a}_Q(V) = \tilde{f}_Q(V) I_p^{\mathrm{T}} - I_p \tilde{f}_Q(V)^{\mathrm{T}}, \tag{9.22}$$

and we observe that $\tilde{a}_Q(V) \in \mathfrak{m}$. Then the reductive SRVT is

$$\mathcal{R}_{\mathfrak{m}}(Y) := \frac{\tilde{a}_Y(\dot{Y})}{\sqrt{\|\tilde{a}_Y(\dot{Y})\|}} = \frac{\tilde{f}_Y(\dot{Y}) I_p^{\mathrm{T}} - I_p \tilde{f}_Y(\dot{Y})^{\mathrm{T}}}{\sqrt{\|\tilde{f}_Y(\dot{Y}) I_p^{\mathrm{T}} - I_p \tilde{f}_Y(\dot{Y})^{\mathrm{T}}\|}}. \tag{9.23}$$

9.4.3 SRVT on the Grassmann manifold: $\mathrm{SO}(n)/(\mathrm{SO}(n-p) \times \mathrm{SO}(p))$

In this section we consider the case when $G = \mathrm{SO}(n)$ and $H = \mathrm{SO}(n-p) \times \mathrm{SO}(p) \subset \mathrm{SO}(n)$ where the elements of $\mathrm{SO}(n-p) \times \mathrm{SO}(p)$ are of the type

$$h = \begin{bmatrix} \Lambda & 0 \\ 0 & \Gamma \end{bmatrix}, \tag{9.24}$$

with Λ a $p \times p$ matrix and Γ an $(n-p) \times (n-p)$ matrix, both orthogonal with determinant equal to 1. We consider the canonical left action of $\mathrm{SO}(n)$ on the quotient $\mathrm{SO}(n)/(\mathrm{SO}(n-p) \times \mathrm{SO}(p))$. This homogeneous manifold can be identified with a quotient of the Stiefel manifold $\mathbb{V}_p(\mathbb{R}^n)/\mathrm{SO}(p)$ with equivalence classes $[Q] = \{\tilde{Q} \in \mathbb{V}_p(\mathbb{R}^n) \mid \tilde{Q} = Q\Lambda,\ \Lambda \in \mathfrak{so}(p)\}$. We denote such a manifold here with $\mathbf{G}_{p,n}(\mathbb{R})$[9]. The reductive subspace is $\mathfrak{m} = (\mathfrak{so}(p) \times \mathfrak{so}(n-p))^{\perp}$ with elements as

[9] An alternative representation of $\mathbf{G}_{p,n}$ is given by considering symmetric matrices P, $n \times n$, with $\mathrm{rank}(P) = p$ and $P^2 = P$, [14].

in (9.20) but with $\Omega = 0$. Imposing a choice of isotropy $B \in \mathfrak{so}(p) \times \mathfrak{so}(n-p)$ such that $(\mathrm{Ad}_{[Q,Q^{\perp}]^T}(A) + B) \in \mathfrak{m}$ leads to the following characterisation of tangent vectors.

Proposition 9.15. Any tangent vector V at $Q \in \mathbf{G}_{p,n}(\mathbb{R})$ is an $n \times p$ matrix such that $Q^T V = 0$, and V can be expressed in the form (9.15) with $F = V$.

The proof follows from (9.12) assuming $g(t) = [Q(t)Q(t)^{\perp}] \in \mathrm{SO}(n)$, and $h(t)$ of the form (9.24), imposing the stated choice of isotropy, and projecting the resulting curves on $\mathbb{V}_p(\mathbb{R}^n)$ by post-multiplication by I_p.

We proceed by using (9.15) but with $F = V$. Define $a_Q : T_Q\mathcal{M} \to \mathfrak{g}$ as in (9.18) with $f_Q : T_Q\mathcal{M} \to T_Q\mathcal{M}$, the identity map $f_Q(V) = V$. Suppose that $Y(t)$ is a curve on the Grassmann manifold, then the SRVT of Y is a curve on $\mathfrak{so}(n)$ and takes the form (9.19) which here becomes

$$\mathcal{R}(Y) := \frac{\dot{Y}Y^{\mathrm{T}} - Y\dot{Y}^{\mathrm{T}}}{\sqrt{\|\dot{Y}Y^{\mathrm{T}} - Y\dot{Y}^{\mathrm{T}}\|}}. \tag{9.25}$$

The reductive SRVT is defined by (9.23) with

$$\tilde{f}_Q(V) = [Q, Q^{\perp}]^T V = \begin{bmatrix} O \\ (Q^{\perp})^T V \end{bmatrix}$$

and $\tilde{a}_Q$ as in (9.22), which implies $\tilde{a}_Q(V) \in \mathfrak{m}$.

9.5 Numerical experiments

To demonstrate an application of the SRVT introduced in this paper, we present a simple example of interpolation between two curves on the unit 2-sphere. In the following we describe some implementation details for this example.

9.5.1 (Preliminaries)**.** We will use Rodrigues' formula for the Lie group exponential,

$$\exp(\hat{x}) = I + \frac{\sin(\alpha)}{\alpha}\hat{x} + \frac{1-\cos(\alpha)}{\alpha^2}\hat{x}^2,$$

with

$$\alpha = \|x\|_2, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \mapsto \hat{x} = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix},$$

where $x \mapsto \hat{x}$ defines an isomorphism between vectors in $\mathbb{R}^3$ and 3×3 skew-symmetric matrices in $\mathfrak{so}(3)$.

9.5.2 (Interpolated curves)**.** Given a continuous curve $c(t), t \in [t_0, t_N]$ on the Stiefel manifold SO(3)/SO(2), which is diffeomorphic to S^2, we replace $c(t)$ with the curve $\bar{c}(t)$ interpolating between $N+1$ values $\bar{c}_i = c(t_i)$, with $t_0 < t_1 < ... < t_N$, as follows:

$$\bar{c}(t) := \sum_{i=0}^{N-1} \chi_{[t_i,t_{i+1})}(t) \exp\left(\frac{t-t_i}{t_{i+1}-t_i}\left(v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}\right)\right)\bar{c}_i, \tag{9.26}$$

where χ is the characteristic function, exp is the Lie group exponential, and v_i are approximations to $\frac{d}{dt}c(t)\Big|_{t=t_i}$ found by solving the equations

$$\bar{c}_{i+1} = \exp\left(v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}\right)\bar{c}_i \tag{9.27}$$

$$\text{constrained by} \quad v_i^{\mathrm{T}}\bar{c}_i = 0. \tag{9.28}$$

The v_i, $i = 1,\ldots,N$, can be found explicitly, by a simple calculation. We observe that if $\kappa = \bar{c}_i \times v_i$, then $\hat{\kappa} = v_i\bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}$. By (9.28), we have that $\|\bar{c}_i \times v_i\|_2 = \|\bar{c}_i\|_2\|v_i\|_2 = \|v_i\|_2$, where the last equality follows because we assume the sphere to have radius 1, and so $\|\bar{c}_i\|_2 = \bar{c}_i^{\mathrm{T}}\bar{c}_i = 1$. Using Rodrigues' formula, from (9.27) we obtain

$$\bar{c}_{i+1} = \frac{\sin(\|v_i\|_2)}{\|v_i\|_2}v_i + \cos\left(\|v_i\|_2\right)\bar{c}_i.$$

Thus $\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1} = 1 - \cos\left(\|v_i\|_2\right)$ and so $\|v_i\|_2 = \arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)$ leading to

$$v_i = \left(\bar{c}_{i+1} - \bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\bar{c}_i\right)\frac{\arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)}{\sqrt{1-\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)^2}}.$$

Inserting this into (9.26) gives

$$\bar{c}(t) = \sum_{i=0}^{N-1} \chi_{[t_i,t_{i+1})}(t) \exp\left(\frac{t-t_i}{t_{i+1}-t_i}\frac{\arccos\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)}{\sqrt{1-\left(\bar{c}_i^{\mathrm{T}}\bar{c}_{i+1}\right)^2}}\left(\bar{c}_{i+1}\bar{c}_i^{\mathrm{T}} - \bar{c}_i\bar{c}_{i+1}^{\mathrm{T}}\right)\right)\bar{c}_i. \tag{9.29}$$

9.5.3 (The SRVT and its inverse)**.** By Definition 9.2 and formulae (9.17), (9.18) and (9.19), the SRVT of the curve (9.29) is a piecewise constant function $\bar{q}(t)$ in $\mathfrak{so}(3)$, taking values $\bar{q}_i = \bar{q}(t_i)$, $i = 0,...,N-1$, where $\bar{q}_i = \mathcal{R}(\bar{c})\big|_{t=t_i}$ is given by

$$\begin{aligned}\bar{q}_i &= \frac{v_i \bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}}}{\| v_i \bar{c}_i^{\mathrm{T}} - \bar{c}_i v_i^{\mathrm{T}} \|^{\frac{1}{2}}} \\ &= \frac{\arccos^{\frac{1}{2}}\left(\bar{c}_i^{\mathrm{T}} \bar{c}_{i+1}\right)}{\left(1-\left(\bar{c}_i^{\mathrm{T}} \bar{c}_{i+1}\right)^2\right)^{\frac{1}{4}} \|\bar{c}_{i+1} \bar{c}_i^{\mathrm{T}} - \bar{c}_i \bar{c}_{i+1}^{\mathrm{T}}\|^{\frac{1}{2}}} \left(\bar{c}_{i+1} \bar{c}_i^{\mathrm{T}} - \bar{c}_i \bar{c}_{i+1}^{\mathrm{T}}\right).\end{aligned} \tag{9.30}$$

Here the norm $\|\cdot\|$ is induced by the negative (scaled) Killing form, which for skew-symmetric matrices corresponds to the Frobenius inner product, $\|A\| = \sqrt{\mathrm{tr}(AA^{\mathrm{T}})}$.

The inverse SRVT is then given by (9.29), with

$$\bar{c}_{i+1} = \exp\left(\|\bar{q}_i\| \bar{q}_i\right) \bar{c}_i, \quad i = 1, \ldots, N-1, \quad \bar{c}_0 = c(t_0).$$

9.5.4 (The reductive SRVT)**.** Since $\mathrm{Evol}(a_{\bar{c}_i}(v_i)) = \exp\left(a_{\bar{c}_i}(v_i)\right)$, the reductive SRVT (9.3.6) becomes then

$$\begin{aligned}\mathcal{R}_{\mathfrak{m}}(\bar{c})\big|_{t=t_i} &= \mathcal{R}(\bar{c})\big|_{t=t_i} \\ &= \frac{\arccos^{\frac{1}{2}}\left(\tilde{c}_i^{\mathrm{T}} \tilde{c}_{i+1}\right)}{\left(1-\left(\tilde{c}_i^{\mathrm{T}} \tilde{c}_{i+1}\right)^2\right)^{\frac{1}{4}} \|\tilde{c}_{i+1} \tilde{c}_i^{\mathrm{T}} - \tilde{c}_i \tilde{c}_{i+1}^{\mathrm{T}}\|^{\frac{1}{2}}} \left(\tilde{c}_{i+1} \tilde{c}_i^{\mathrm{T}} - \tilde{c}_i \tilde{c}_{i+1}^{\mathrm{T}}\right),\end{aligned} \tag{9.31}$$

with

$$\begin{aligned}&\tilde{c}_i = [U, U^{\perp}]_i^{\mathrm{T}} \bar{c}, \quad i = 0, \ldots, N, \\ &[U, U^{\perp}]_{i+1} = \exp\left(a_{\bar{c}_i}(v_i)\right) [U, U^{\perp}]_i \quad i = 0, \ldots, N-1,\end{aligned}$$

where $[U, U^{\perp}]_0$ can be found e.g. by QR-factorization of $c(t_0)$.

9.5.5 (Curve blending on the 2-sphere)**.** We wish to compute the geodesic in the shape space of curves on the sphere between the two curves $\bar{c}^1(t)$ and $\bar{c}^2(t)$. Following [9], we use a dynamic programming algorithm to solve the optimization problem (9.7) (see [7, 24] for a detailed description on the use of dynamic programming for shapes):

Algorithm 9.1. REPARAMETRISATION [7, Section 3.2]

```
Given $\bar{q}^1(t), \bar{q}^2(t), N, \{t_i\}_{i=0}^N$
for $i, j \in \{0, \ldots, N\}$ do
    $c_{\min} = \infty$
    for $k \in \{0, \ldots, i-1\}, l \in \{0, \ldots, j-1\}$ do
        $c_{\mathrm{loc}} = \int_{t_i}^{t_k} |\bar{q}^1(t) - \bar{q}^2(t_l + \frac{t_j - t_l}{t_i - t_k} t)|^2 \mathrm{d}t$
        **if** $\Psi^m(k,l) = \Psi \circ \cdots \circ \Psi(k,l) = (0,0)$ for some $m \geq 0$ **then**
            $z = 0$
        **else**
            $z = \infty$
        $c = c_{\text{loc}} + A_{k,l} + z$
        **if** $c < c_{\min}$ **then**
            $c_{\min} = c$
            $\Psi(i,j) = (k,l)$
    $A_{i,j} = c_{\min}$
Create two vectors of indices $(p,q)$ by setting $(p_0,q_0) = (N,N)$ and
$(p_{m+1},q_{m+1}) = \Psi(p_m,q_m)$ until $(p_{m+1},q_{m+1}) = (0,0)$
Flip $(p,q)$ so it starts at $(0,0)$ and ends at $(N,N)$
**for** $i \in \{0,...,N\}$ **do**
    $s_i = t_{q_j} + (t_{q_{j+1}} - t_{q_j})\frac{t_i - t_{p_j}}{t_{p_{j+1}} - t_{p_j}}$ for $j$ s.t. $p_j \leq i < p_{j+1}$
Return $s = \{s_i\}_{i=0}^N$
```

With this approach, we reparametrise optimally the curve $\bar{c}^2(t)$ while minimizing its distance to $\bar{c}^1(t)$. This distance is measured by taking the L^2 norm of $\bar{q}^1(t) - \bar{q}^2(t)$ in the Lie algebra. In the discrete case, this reparametrisation yields an optimal set of grid points $\{s_i\}_{i=0}^N$, where $s_0 = t_0 < s_1 < ... < s_N = t_N$, from which we find $\bar{c}_i'^2 = \bar{c}^2(s_i)$ by (9.29). See [9] for further details.

We interpolate between $\bar{c}^1(t)$ and $\bar{c}'^2(t)$ by performing a linear convex combination of their SRV transforms $\bar{q}^1(t)$ and $\bar{q}'^2(t)$, and then by taking the inverse SRVT of the result. We obtain

$$\bar{c}_{\text{int}}(\bar{c}_1, \bar{c}_2', \theta) = \mathcal{R}^{-1}\left(\left(1-\theta\right)\mathcal{R}(\bar{c}_1) + \theta\,\mathcal{R}(\bar{c}_2')\right), \qquad \theta \in [0,1].$$

Examples are reported in Figures 9.2, 9.3 and 9.4, where interpolation between two curves is performed with and without reparametrisation. We show curves resulting from using both (9.30) and (9.31), and compare these to the results obtained when employing the SRVT introduced in [9] on curves in SO(3) which are then traced out by a vector in $\mathbb{R}^3$ to match the curves in S^2 studied here.

9.5.6 (Conclusions)**.** We have proposed generalisations of the SRVT approach to curves and shapes evolving on homogenous manifolds using Lie group actions. Different Lie group actions lead to different Riemannian metrics in the infinite dimensional manifolds of curves and shapes opening up for a variety of possibilities which can all be implemented in the same generalised SRVT framework. We have presented only a few preliminary examples here, and further tests and analysis will be the subject of future work. A number of open questions related to the properties of the pullback metrics through the SRVT, to the performance of the algorithms when using different Lie group actions, to the comparison of the SRVT and the reductive SRVT and to the implementation of the approach in examples of non reductive homogeneous manifolds will be addressed in future research.

9.6 Acknowledgment

This work was supported by the Norwegian Research Council, and by the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie, grant agreement No. 691070.

9.7 Appendix: Auxiliary results for Section 9.3

9.7.1 (Auxiliary results for Section 9.3)**.**

Lemma 9.16. For the homogeneous space $\mathcal{M} = G/H$ with projection $\pi\colon G \to G/H$ the derivation map $D_{\mathcal{M}}\colon C^\infty(I, G/H) \to C^\infty(I, T(G/H)), c \mapsto \dot{c}$ is smooth.

Proof. The map $D_G\colon C^\infty(I,G) \to C^\infty(I,TG),\ \gamma \mapsto \dot{\gamma}$ is a smooth group homomorphism by [13, Lemma 2.1]. As $\pi\colon G \to G/H$ is a smooth submersion, $\theta_\pi\colon C^\infty(I,G) \to C^\infty(I,G/H), c \mapsto \pi \circ c$ is a smooth submersion [2, Lemma 2.4]. Write $\theta_{T\pi} \circ D_G = D \circ \theta_\pi$, whence by [12, Lemma 1.8] $D_{\mathcal{M}}$ is smooth. □

Lemma 9.17. With $\theta := \theta_\omega \circ D$ The identity (9.10) $\mathrm{id}_{C^\infty_{eH}(I,\mathcal{M})} = \pi \circ \mathrm{Evol} \circ \theta$ holds.

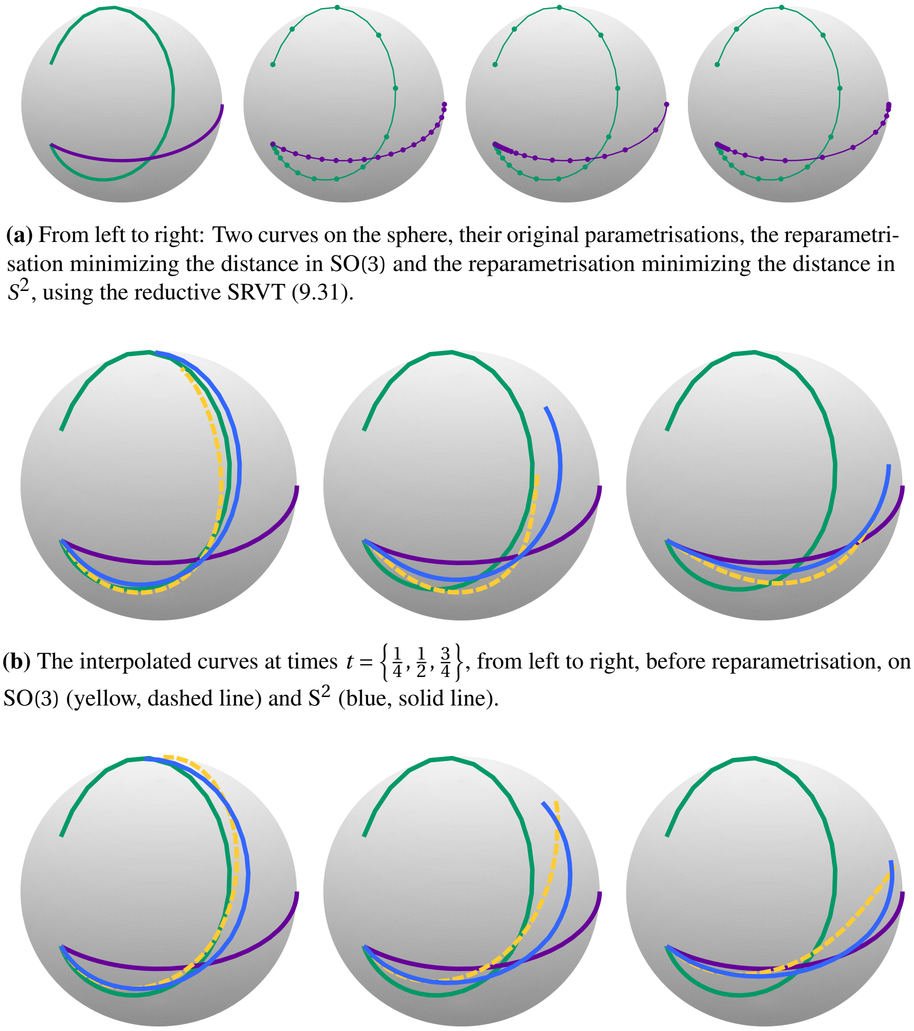
Proof. Let $c\colon I \to \mathcal{M}$ be smooth with $c(0) = eH$ and choose $g\colon I \to G$ smooth with $g(0) = e$ and $\pi \circ g = c$. Set $\gamma(t) := \mathrm{Evol}(\theta(c))(t)$. It suffices to prove that $\gamma(t)^{-1}g(t) \in H$ for all $t \in I$. Then $\pi \circ \gamma = \pi \circ g = c$ and the assertion follows.

As $\gamma(0)^{-1}g(0) = e \in H$, we only have to prove that $\frac{\mathrm{d}}{\mathrm{d}t}\pi(\gamma(t)^{-1}g(t))$ vanishes everywhere to obtain $\gamma(t)^{-1}g(t) \in H$. Before we compute the derivative of $\pi(\gamma(t)^{-1}g(t))$, let us first collect some facts concerning the logarithmic derivatives $\delta^r(f) = \dot{f}.f^{-1}$ and $\delta^l(f) = f^{-1}.\dot{f}$. By definition $\delta^r(\gamma) = \delta^r(\mathrm{Evol}(\theta(c))) = \theta(c)$. Further, [17, Lemma 38.1] yields for smooth $f, h\colon I \to G$:

$$\delta^r(f \cdot h) = \delta^r(f) + \mathrm{Ad}(f).\delta^r(h) \quad \text{and} \quad \delta^r(f^{-1}) = -\delta^l(f), \tag{9.32}$$

whence

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(\gamma(t)^{-1}g(t)) &= (\gamma(t)^{-1}g(t)) \cdot \delta^l(\gamma^{-1}g)(t) \\ &\overset{(9.32)}{=} -(\gamma(t)^{-1}g(t)) \cdot \delta^r(g^{-1}\gamma)(t) \\ &\overset{(9.32)}{=} (\gamma(t)^{-1}g(t)) \cdot (\delta^l(g)(t) - \mathrm{Ad}(g(t)^{-1}).\theta(c)(t))\end{aligned}$$

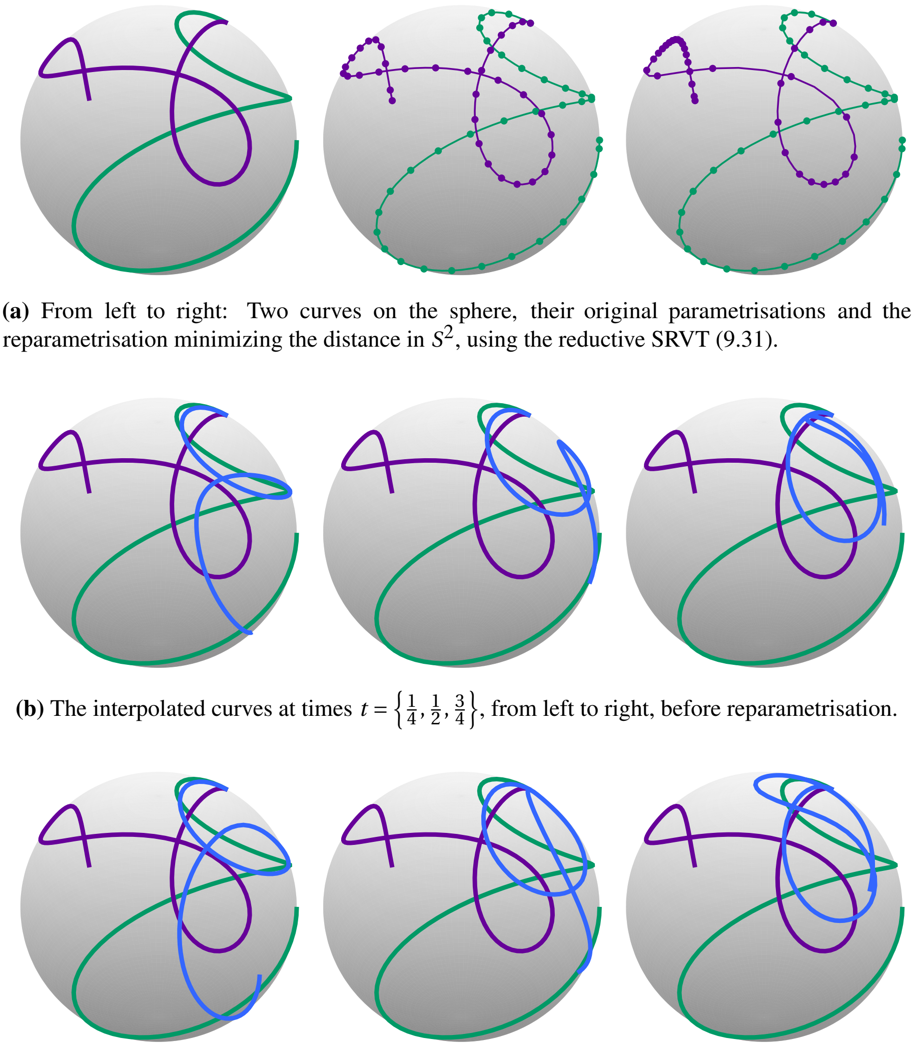


(a) From left to right: Two curves on the sphere, their original parametrisations, the reparametrisation minimizing the distance in SO(3) and the reparametrisation minimizing the distance in S^2, using the reductive SRVT (9.31).

(b) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, before reparametrisation, on SO(3) (yellow, dashed line) and S^2 (blue, solid line).

(c) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, after reparametrisation, on SO(3) (yellow, dashed line) and S^2 (blue, solid line).

Figure 9.2: Interpolation between two curves on S^2, with and without reparametrisation, obtained by the reductive SRVT (9.31). The results obtained by using the SRVT (9.30) are not identical to these, but in this case very similar, and therefore omitted. The results are compared to the corresponding SRVT interpolation between curves on SO(3), which are then mapped to S^2 by multiplying with the vector $(1,0,0)^{\mathrm{T}}$. The curves are $c^1(t) = R_x(\pi t^3)R_y(\pi t^3)R_y(\pi t^3/2)\cdot(1,0,0)^{\mathrm{T}}$ and $c^2(t) = R_z(3\pi t/4)R_x(\pi t)\cdot(1,0,0)^{\mathrm{T}}$ for $t \in [0,1]$, where $R_x(t)$, $R_y(t)$ and $R_z(t)$ are the rotation matrices in SO(3) corresponding to rotation of an angle t around the x-, y- and z-axis, respectively.

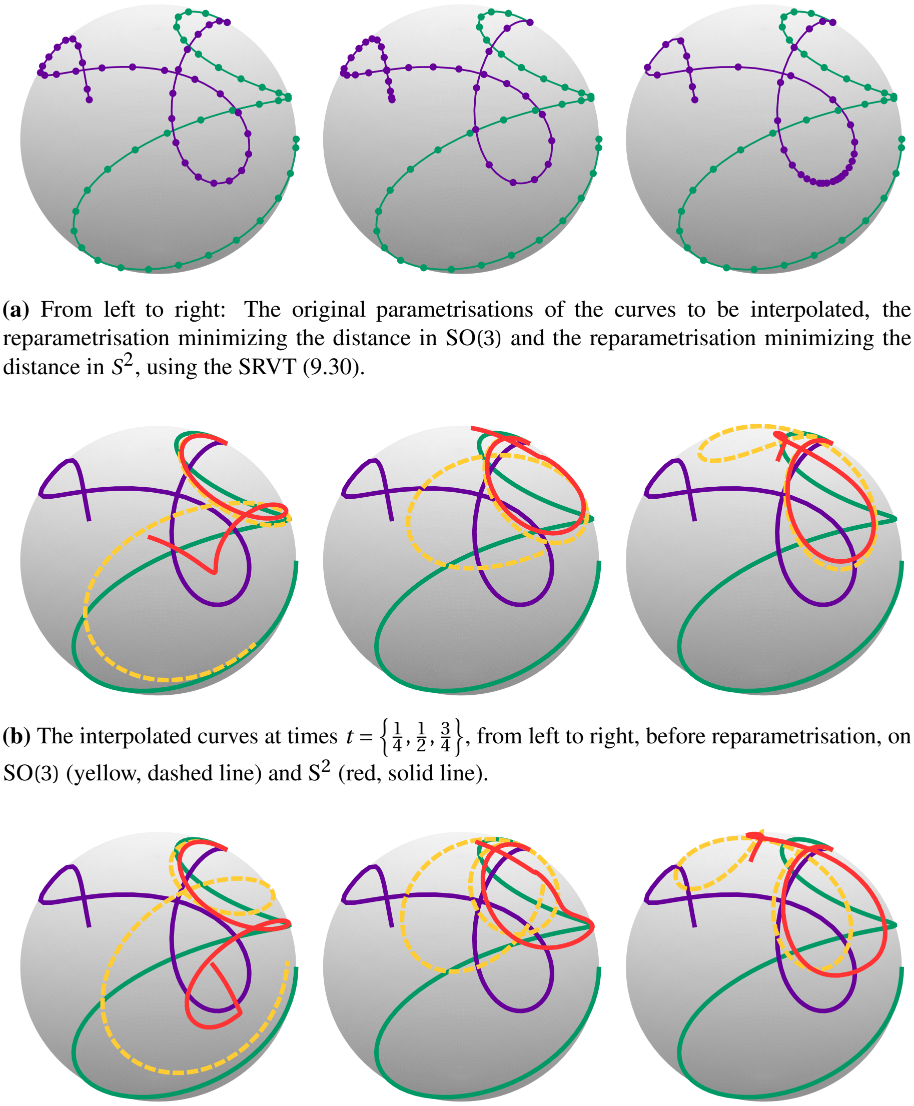


(a) From left to right: Two curves on the sphere, their original parametrisations and the reparametrisation minimizing the distance in S^2, using the reductive SRVT (9.31).

(b) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, before reparametrisation.

(c) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, after reparametrisation.

Figure 9.3: Interpolation between two curves on S^2, with and without reparametrisation, found by the reductive SRVT (9.31). The curves are $c^1(t) = R_x(2\pi t)R_y(2\pi t)R_z(\pi t) \cdot (0,1,1)^{\mathrm{T}}/\sqrt{2}$ and $c^2(t) = R_z(2\pi t)R_x(2\pi t)R_y(\pi t/2) \cdot (0,1,1)^{\mathrm{T}}/\sqrt{2}$ for $t \in [0,1]$, where $R_x(t)$, $R_y(t)$ and $R_z(t)$ are the rotation matrices in SO(3) corresponding to rotation of an angle t around the x-, y- and z-axis, respectively.



(a) From left to right: The original parametrisations of the curves to be interpolated, the reparametrisation minimizing the distance in SO(3) and the reparametrisation minimizing the distance in S^2, using the SRVT (9.30).

(b) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, before reparametrisation, on SO(3) (yellow, dashed line) and S^2 (red, solid line).

(c) The interpolated curves at times $t = \left\{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}\right\}$, from left to right, after reparametrisation, on SO(3) (yellow, dashed line) and S^2 (red, solid line).

Figure 9.4: Interpolation between the same curves as in Figure 9.3, with and without reparametrisation, obtained here with the SRVT (9.30), compared to the corresponding interpolation between curves on SO(3) mapped to S^2 by multiplication with the vector $(0,1,1)^{\mathrm{T}}/\sqrt{2}$.

Recall that by definition, $\theta(c)(t) = \omega(\dot{c}(t)) = \mathrm{Ad}(g(t)).\omega_e(T\Lambda^{g(t)^{-1}}(\dot{c}(t)))$ (here $\pi \circ g = c$ is used). Inserting this into the above equation we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}(\gamma(t)^{-1}g(t)) = (\gamma(t)^{-1}g(t)) \cdot (\delta^l(g)(t) - \omega_e(T\Lambda^{g(t)^{-1}} \circ \dot{c}(t))). \tag{9.33}$$

Observe that $T_e\pi(\delta^l(g)(t)) = T\Lambda^{g(t)^{-1}} T\pi\dot{g}(t) = T\Lambda^{g(t)^{-1}}\dot{c}(t)$ since $\pi \circ g = c$. As ω_e is a section of $T_e\pi$, $T_e\pi(\delta^l(g)(t) - \omega_e(T\Lambda^{g(t)^{-1}} \circ \dot{c}(t))) = 0 \in T_{eH}\mathcal{M}$. Summing up

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\pi(\gamma(t)^{-1}g(t)) &\overset{(9.33)}{=} T\pi((\gamma(t)^{-1}g(t)) \cdot (\delta^l(g)(t) - \omega_e(T\Lambda^{g(t)^{-1}} \circ \dot{c}(t))) \\ &\overset{(9.8)}{=} T\Lambda^{\gamma(t)^{-1}g(t)} T_e\pi(\delta^l(g)(t) - \omega_e(T\Lambda^{g(t)^{-1}} \circ \dot{c}(t))) = 0. \square\end{aligned}$$

9.7.2 (A chart for the image of the SRVT)**.** Let G be a Lie group with Lie algebra $\mathfrak{g}$. Using the adjoint action of G on $\mathfrak{g}$ and the evolution $\mathrm{Evol}\colon C^\infty(I,\mathfrak{g}) \to C^\infty(I,G)$, we define the map

$$\Psi\colon C^\infty(I,\mathfrak{g}) \to C^\infty(I,\mathfrak{g}), \quad q \mapsto -\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q,$$

where the dot denotes pointwise application of the linear map $\mathrm{Ad}(\mathrm{Evol}(q)^{-1})$. Observe that Ψ (co)restricts to a mapping $C^\infty(I,\mathfrak{g}\setminus\{0\}) \to C^\infty(I,\mathfrak{g}\setminus\{0\})$.

Lemma 9.18. The map $\Psi\colon C^\infty(I,\mathfrak{g}) \to C^\infty(I,\mathfrak{g})$ is a smooth involution.

Proof. To establish smoothness of Ψ, consider the commutative diagram

$$\begin{tikzcd} C^\infty(I,\mathfrak{g}) \arrow[r, "\Psi"] \arrow[d, "(\mathrm{Evol},\mathrm{id}_{C^\infty(I,\mathfrak{g})})"'] & C^\infty(I,\mathfrak{g}) \arrow[d, equal] \\ C^\infty(I,G) \times C^\infty(I,\mathfrak{g}) \arrow[r, "(f,g)\mapsto \mathrm{Ad}(f).g"] & C^\infty(I,\mathfrak{g}) \end{tikzcd}$$

As $\mathrm{Ad}\colon G\times\mathfrak{g} \to \mathfrak{g}$ is smooth, so is $(f,g) \mapsto \mathrm{Ad}(f).g$ (cf. [13, Proof of Proposition 6.2]) and Ψ is smooth as a composition of smooth maps. Compute for $q \in C^\infty(I,\mathfrak{g})$

$$\begin{aligned}\Psi(\Psi(q)) &= -\mathrm{Ad}(\mathrm{Evol}(\Psi(q))^{-1}).\Psi(q) \\ &= -\mathrm{Ad}(\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q)^{-1}).(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q) \\ &= \mathrm{Ad}((\mathrm{Evol}(q)\,\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q))^{-1}).q.\end{aligned}$$

To see that $\Psi(\Psi(q)) = q$, we prove that $\gamma_q := \mathrm{Evol}(q)\,\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q)$ is a constant path. Recall that $\mathrm{Evol}(q)$ and $\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q)$ are smooth paths starting at the identity in G. Hence it suffices to prove $\delta^r(\gamma_q) = 0$. To this end, apply the product formula (9.32) and $\delta^r(\mathrm{Evol}(q)) = q$:

$$\begin{aligned}\delta^r(\gamma_q)) &= \delta^r(\mathrm{Evol}(q)) + \mathrm{Ad}(\mathrm{Evol}(q)).\delta^r(\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q)) \\ &= q + \mathrm{Ad}(\mathrm{Evol}(q)).(-\mathrm{Ad}(\mathrm{Evol}(q)^{-1}).q) = q - q = 0. \qquad \square\end{aligned}$$

To account for the initial point $c_0 \in \mathcal{M}$, fix $g_0 \in \pi^{-1}(c_0)$ and define

$$\Psi_{g_0}\colon C^\infty(I,\mathfrak{g}) \to C^\infty(I,\mathfrak{g}), \quad \Psi_{g_0}(q) := \mathrm{Ad}(g_0).\Psi(q) = -\mathrm{Ad}(g_0\,\mathrm{Evol}(q)^{-1}).q.$$

For k in the center of G, $\Psi_k = \Psi$ holds, but in general Ψ_{g_0} will not be an involution.

Lemma 9.19. For each $g_0 \in G$, the map Ψ_{g_0} is a diffeomorphism with inverse $\Psi_{g_0^{-1}}$.

Proof. From the definition of Ψ_{g_0} and Lemma 9.18, it is clear that Ψ_{g_0} is a smooth diffeomorphism. We use that $\mathrm{Ad}\colon G \to \mathrm{GL}(\mathfrak{g})$ is a group morphism and compute

$$\begin{aligned}\Psi_{g_0^{-1}}(\Psi_{g_0}(q)) &= \mathrm{Ad}(g_0^{-1}).\Psi(\Psi_{g_0}(q)) = \mathrm{Ad}(g_0).\Psi(\mathrm{Ad}(g_0).\Psi(q))\\ &= \mathrm{Ad}(g_0^{-1}).\Big(-\mathrm{Ad}(\mathrm{Evol}(\mathrm{Ad}(g_0).\Psi(q)))^{-1}).\mathrm{Ad}(g_0).\Psi(q)\Big)\\ &= -\mathrm{Ad}(g_0^{-1}g_0\,\mathrm{Evol}(\Psi(q))^{-1}g_0^{-1}g_0).\Psi(q) = \Psi(\Psi(q)) = q.\end{aligned}$$

Here we used that $\mathrm{Evol}(\mathrm{Ad}(g).f) = g\,\mathrm{Evol}(f)g^{-1}$, for $g \in G$. □

Lemma 9.20. Fix $c_0 \in \mathcal{M}$ and choose $g_0 \in G$ with $\pi(g_0) = c_0$. Assume that $\mathcal{M}$ is reductive with $\mathfrak{g} = \mathfrak{h} \oplus \mathfrak{m}$, then

$$\Psi_{g_0}(C^\infty(I,\mathfrak{m}\setminus\{0\})) = \{f \in C^\infty(I,\mathfrak{g}) \mid f = \theta_\omega(\dot c) \quad \text{for some } c \in \mathrm{Imm}_{c_0}(I,\mathcal{M})\}.$$

With θ_ω as in 9.3.4 the formula $\theta_\omega \circ D(\rho_{c_0} \circ \Psi_{g_0}(q)) = \Psi_{g_0}(q)$ holds.

Proof. Consider $c \in \mathrm{Imm}_{c_0}(I,\mathcal{M})$ and recall from Proposition 9.9 the identity $\Lambda_{c_0}(\mathrm{Evol}(\theta_\omega(\dot c)) = \pi(\mathrm{Evol}(\theta_\omega(\dot c))g_0) = c$. Choose $\hat c = \mathrm{Evol}(\theta_\omega(\dot c))g_0$ as a smooth lift of c to G and compute as follows:

$$\begin{aligned}\Psi_{g_0^{-1}}(\theta_\omega(\dot c)) &= \mathrm{Ad}(g_0^{-1}).\Big(-\mathrm{Ad}(\mathrm{Evol}(\theta_\omega(\dot c))^{-1}).(\theta_\omega(\dot c))\Big)\\ &= \mathrm{Ad}(g_0^{-1}).\Big(-\mathrm{Ad}(\mathrm{Evol}(\theta_\omega(\dot c))^{-1}).\mathrm{Ad}(\hat c).\omega_e(T\Lambda^{\hat c^{-1}}(\dot c))\Big)\\ &= \mathrm{Ad}(g_0^{-1}).\Big(-\mathrm{Ad}(\mathrm{Evol}(\theta_\omega(\dot c))^{-1}).\mathrm{Ad}(\mathrm{Evol}(\theta_\omega(\dot c))g_0).\omega_e(T\Lambda^{\hat c^{-1}}(\dot c))\Big)\\ &= -\omega_e(T\Lambda^{\hat c^{-1}}(\dot c)) \in \mathfrak{m}\setminus\{0\}.\end{aligned}$$

Conversely, let us show that $\Psi_{g_0}(C^\infty(I,\mathfrak{m}\setminus\{0\}))$ is contained in the image of $\theta_\omega \circ D|_{\mathrm{Imm}_{c_0}(I,\mathcal{M})}$. To this end, consider $q = \Psi_{g_0}(v)$ for $v \in C^\infty(I,\mathfrak{m}\setminus\{0\})$. We compute

$$\begin{aligned}\rho_{c_0}(q) &= \Lambda_{c_0}(\mathrm{Evol}(\mathrm{Ad}(g_0).\Psi(v))) = \pi(\mathrm{Evol}(\mathrm{Ad}(g_0).\Psi(v)g_0))\\ &= \pi(g_0\,\mathrm{Evol}(\Psi(v))) = \pi(g_0\,\mathrm{Evol}(-\mathrm{Ad}(\mathrm{Evol}(v)^{-1}).v)) \qquad (9.34)\\ &= \Lambda^{g_0}(\pi(\mathrm{Evol}(\Psi(v)))).\end{aligned}$$

Since Λ^{g_0} is a diffeomorphism, $\rho_{c_0}(q)\colon I \to \mathcal{M}$ is an immersion if and only if the curve $\pi(\mathrm{Evol}(\Psi(v)))$ has a non-vanishing derivative everywhere. Recall from the proof of Lemma 9.18 that $\mathrm{Evol}(v)\,\mathrm{Evol}(\Psi(v)) = e$, whence we compute the derivative

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\pi(\mathrm{Evol}(\Psi(v))(t)) &= T\pi\left(\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Evol}(\Psi(v))(t)\right)\\ &= T\pi(\Psi(v)\,\mathrm{Evol}(\Psi(v)))(t)\\ &= T\pi(-\mathrm{Ad}(\mathrm{Evol}(v)^{-1}).v\,\mathrm{Evol}(\Psi(v)))(t)\\ &= -T\pi\circ(L_{\mathrm{Evol}(v)^{-1}(t)})_*\circ(R_{\mathrm{Evol}(v)(t)\,\mathrm{Evol}(\Psi(v))(t)})_*(v(t))\\ &= -T\Lambda^{\mathrm{Evol}(v)^{-1}(t)}\circ T_e\pi(v(t)).\end{aligned} \tag{9.35}$$

In passing to the last line, we used that π commutes with the left action. Since $T\Lambda^g$ is an isomorphism, $\frac{\mathrm{d}}{\mathrm{d}t}\pi(\mathrm{Evol}(\Psi(v))(t))$ vanishes if and only if $v(t) \in \ker T_e\pi = \mathfrak{h}$. However, $v(t) \in \mathfrak{m}\setminus\{0\}$, whence $\rho_{c_0}(\Psi_{g_0}(v)) \in \mathrm{Imm}_{c_0}(I,\mathcal{M})$ and we can apply $\theta_\omega\circ D$ to $\rho_{c_0}(q)$. A combination of (9.34) and (9.35) yields $\frac{\mathrm{d}}{\mathrm{d}t}\rho_{c_0}(\Psi_{g_0}(v))(t) = -T\Lambda^{g_0(\mathrm{Evol}(v))^{-1}(t)}\circ T_e\pi(v(t))$. With

$$\pi(g_0\,\mathrm{Evol}(v)^{-1}) = \rho_{c_0}(\Psi_{g_0}(v)(t)),$$

this yields

$$\begin{aligned}&\theta_\omega\left(\frac{\mathrm{d}}{\mathrm{d}t}\rho_{c_0}(v(t))\right)=\theta_\omega\left(\frac{\mathrm{d}}{\mathrm{d}t}\rho_{c_0}(\Psi_{g_0}(v))(t)\right)=\theta_\omega(-T\Lambda^{g_0(\mathrm{Evol}(v))^{-1}(t)}\circ T_e\pi(v(t)))\\ &=\mathrm{Ad}(g_0(\mathrm{Evol}(v))^{-1}).\omega_e(-T\Lambda^{(g_0(\mathrm{Evol}(v))^{-1}(t))^{-1}}T\Lambda^{g_0(\mathrm{Evol}(v))^{-1}(t)}\circ T_e\pi(v(t)))\\ &=-\mathrm{Ad}(g_0(\mathrm{Evol}(v))^{-1}).\omega_e(T_e\pi(v(t)) = -\mathrm{Ad}(g_0(\mathrm{Evol}(v))^{-1}).v(t) = \Psi_{g_0}(v)(t).\end{aligned}$$

Note that as $\omega_e = (T_e\pi|_{\mathfrak{m}})^{-1}$, we have $\omega_e(T_e\pi(v(t)) = v(t)$. □

Bibliography

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2007.

[2] H. Amiri and A. Schmeding. A differentiable monoid of smooth maps on Lie groupoids. *J. Lie Theory*, 29(4):1167–1192, 2019.

[3] A. Bastiani. Applications différentiables et variétés différentiables de dimension infinie. *J. Analyse Math.*, 13:1–114, 1964.

[4] M. Bauer, M. Bruveris, P. Harms, and P. W. Michor. Vanishing geodesic distance for the Riemannian metric with geodesic equation the KdV-equation. *Ann. Global Anal. Geom.*, 41(4):461–472, 2012.

[5] M. Bauer, M. Bruveris, S. Marsland, and P. W. Michor. Constructing reparameterization invariant metrics on spaces of plane curves. *Differential Geom. Appl.*, 34:139–165, 2014.

[6] M. Bauer, M. Bruveris, and P. W. Michor. Overview of the geometries of shape spaces and diffeomorphism groups. *J. Math. Imaging Vision*, 50(1-2):60–97, 2014.

[7] M. Bauer, M. Eslitzbichler, and M. Grasmair. Landmark-guided elastic shape analysis of human character motions. *Inverse Probl. Imaging*, 11(4):601–621, 2017.

[8] M. Bruveris. Optimal reparametrizations in the square root velocity framework. *SIAM J. Math. Anal.*, 48(6):4335–4354, 2016.

[9] E. Celledoni, M. Eslitzbichler, and A. Schmeding. Shape analysis on Lie groups with applications in computer animation. *J. Geom. Mech.*, 8(3):273–304, 2016.

[10] E. Celledoni and B. Owren. On the implementation of Lie group methods on the Stiefel manifold. *Numer. Algorithms*, 32(2-4):163–183, 2003.

[11] S. Gallot, D. Hulin, and J. Lafontaine. *Riemannian geometry*. Universitext. Springer-Verlag, Berlin, third edition, 2004.

[12] H. Glöckner. Fundamentals of submersions and immersions between infinite-dimensional manifolds, Mar. 2015. arXiv:1502.05795v3.

[13] H. Glöckner. Regularity properties of infinite-dimensional Lie groups, and semiregularity, 2015. arXiv:1208.0715v3.

[14] K. Hüper and F. Silva Leite. On the geometry of rolling and interpolation curves on S^n, SO_n, and Grassmann manifolds. *J. Dyn. Control Syst.*, 13(4):467–502, 2007.

[15] A. W. Knapp. *Lie groups beyond an introduction*, volume 140 of *Progress in Mathematics*. Birkhäuser Boston, Inc., Boston, MA, second edition, 2002.

[16] S. Kobayashi and K. Nomizu. *Foundations of differential geometry. Vol. II*. Interscience Tracts in Pure and Applied Mathematics, No. 15 Vol. II. Interscience Publishers John Wiley & Sons, Inc., New York-London-Sydney, 1969.

[17] A. Kriegl and P. W. Michor. *The convenient setting of global analysis*, volume 53 of *Mathematical Surveys and Monographs*. American Mathematical Society, Providence, RI, 1997.

[18] A. Le Brigant. Computing distances and geodesics between manifold-valued curves in the SRV framework. *J. Geom. Mech.*, 9(2):131–156, 2017.

[19] P. W. Michor. *Manifolds of differentiable mappings*, volume 3 of *Shiva Mathematics Series*. Shiva Publishing Ltd., Nantwich, 1980.

[20] P. W. Michor and D. Mumford. Vanishing geodesic distance on spaces of submanifolds and diffeomorphisms. *Doc. Math.*, 10:217–245, 2005.

[21] P. W. Michor and D. Mumford. Riemannian geometries on spaces of plane curves. *J. Eur. Math. Soc. (JEMS)*, 8(1):1–48, 2006.

[22] H. Munthe-Kaas and O. Verdier. Integrators on homogeneous spaces: isotropy choice and connections. *Found. Comput. Math.*, 16(4):899–939, 2016.

[23] J.-P. Ortega and T. S. Ratiu. *Momentum maps and Hamiltonian reduction*, volume 222 of *Progress in Mathematics*. Birkhäuser Boston, Inc., Boston, MA, 2004.

[24] T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, Jan 2003.

[25] R. W. Sharpe. *Differential geometry*, volume 166 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1997. Cartan's generalization of Klein's Erlangen program, With a foreword by S. S. Chern.

[26] A. Srivastava, E. Klassen, S. Joshi, and I. Jermyn. Shape analysis of elastic curves in Euclidean spaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33:1415–1428, 2011.

[27] J. Su, S. Kurtek, E. Klassen, and A. Srivastava. Statistical analysis of trajectories on Riemmannian manifolds: bird migration, hurricane tracking and video surveillance. *The Annals of Applied Statistics*, 8(2):530–552, 2014.

[28] Z. Su, E. Klassen, and M. Bauer. The square root velocity framework for curves in a homogeneous space. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 680–689. IEEE, 2017.

[29] Z. Su, E. Klassen, and M. Bauer. Comparing curves in homogeneous spaces. *Differential Geom. Appl.*, 60:9–32, 2018.