universität
wien

Habilitationsschrift

Titel der Habilitationsschrift

Geometric modeling in geophysical fluid dynamics

Verfasser

Dr. Alexander Bihlo

Fachgebiet

Angewandte Mathematik

St. John's, NL (Canada), im Dezember 2018

Abstract

This thesis is devoted to the problem of finding geometrically inspired discrete approximations to the equations of geophysical fluid dynamics. It is well known that these equations admit a rich geometric structure, which can include the invariance under specific transformation groups, numerous conservation laws, and a symplectic or Hamiltonian structure, just to name a few. Unfortunately, these geometric properties are typically lost when discretizing these equations using standard numerical methods. Specialized numerical integrators, so-called geometric numerical integrators, are needed.

We focus on the use of Lie symmetries and conservation laws for constructing tailored parameterization and discretization schemes for some of the important models of (geophysical) fluid dynamics.

In particular, we construct invariant discretization schemes for Burgers' equation, the Korteweg–de Vries equation and the shallow-water equations. A main characteristic of these invariant schemes is that they require the use of adaptive moving meshes. Developing suitable strategies for invariantly adapting moving meshes is hence a cornerstone for these integrators. While most invariant integrators proposed so far are based on the finite difference methodology, we also hint on using finite volume and meshless methods.

Parameterization is the procedure of approximately including processes into numerical models that said models cannot resolve explicitly. Constructing parameterization schemes is a task of immense practical relevance and one of the most active research directions in modern weather and climate prediction. We propose several general methods for constructing parameterization schemes that retain selected symmetries and conservation laws of their parent models. Particular invariant and conservative parameterization schemes are constructed for the barotropic vorticity equation on the beta-plane and the shallow-water equations.

Zusammenfassung

Diese Schrift widmet sich des Problems geometrisch inspirierter diskreter Approximationen der Gleichungen der geophysikalischen Fluiddynamik zu finden. Es ist wohlbekannt dass diese Gleichungen reichhaltige geometrische Strukturen besitzen, die unter anderem die Invarianz unter speziellen Transformationsgruppen, zahlreiche Erhaltungssätze und eine symplektische oder Hamiltonsche Struktur beinalten, um nur wenige Beispiele zu nennen. Unglücklicherweise bleiben diese Strukturen typischerweise nicht erhalten wenn diese Gleichungen mit traditionellen numerischen Verfahren diskretisiert werden. Spezielle numerische Verfahren, sogenannten geometrischen numerischen Integratoren werden benötigt.

Wir konzentrieren uns auf die Verwendung von Lie-Symmetrien und Erhaltungssätzen für die Konstruktion spezieller Parameterisierungen und Diskretisierungen für einige wichtige Modelle der (geophysikalischen) Fluiddynamik.

Im Besonderen konstruieren wir invariante Diskretisierungen für Burgers' Gleichung, die Korteweg–de Vries Gleichung und die Flachwassergleichungen. Eine umfassende Charakteristik dieser invarianten Verfahren ist die Notwendigkeit adaptive Gitter zu verwenden. Wir entwickeln verschiedene Strategien für invariante adaptive Gitterverfahren. Obwohl die meisten invarianten Integratoren auf Finiten Differenzenverfahren basieren geben wir doch Beispiele für die Verwendung von Finiten Volumen und Finiten Elementen.

Parameterisierung widmet sich Annäherungen für unaufgelöste Prozesse zu finden, die in numerische Modelle eingebaut werden können. Die Entwicklung von Parameterisierungen ist eine Aufgabe großer praktischer Wichtigkeit un eine der aktivsten Forschungsrichtungen der modernen Wetter- und Klimavorhersage. Wir schlagen zahlreiche allgemeine Methoden zur Konstruktion invarianter und konservativer Parameterisierungen vor. Als Beispiel werden invariante und konservative Parameterisierungen für die barotrope Vorticitygleichung auf der beta-Ebene und die Flachwassergleichungen konstruiert.

Contents

Chapter 1

Preface

This monograph serves as my habilitation thesis at the University of Vienna, submitted in September 2018. It is based on a selection of my work in the general area of geometric numerical integration with application to geophysical fluid dynamics. The following Sections 1.1 and 1.2 present an overarching introduction and motivation for this work, as well as a guide through the thesis.

1.1 Motivation

Geophysical fluid dynamics deals with the motion of fluids on rotating planets, in particular on the planet Earth. The governing equations of these fluids are the governing equations of hydro-thermodynamics in a rotating reference frame. These equations are intrinsically nonlinear and hence can in general not be solved using analytical methods. Research on geophysical fluid dynamics is hence strongly driven by numerical investigations.

A sub-field of geophysical fluid dynamics of immense practical relevance is the problem of weather and climate prediction. While there are many facets to this problem, from the point of view of solving the underlying governing partial differential equations numerically the main two problems are *discretization* and *parameterization*.

Numerically solving partial differential equations requires to chose a suitable discretization strategy, using e.g. finite differences, finite elements, finite volumes or spectral methods. Irrespective of the discretization methodology chosen, every numerical method has intrinsically limited spatial and temporal resolutions. Since the processes described by the governing equations of geophysical fluid dynamics happen on vastly different scales, ranging from millimetres to thousands of kilometres in space and milliseconds to years in time, resolving all processes explicitly in a numerical model is computationally impossible. Therefore, one has to choose, based on computational resources available, the resolution of the numerical method at hand. This determines the overall grid resolution and thus the specific processes that can and that cannot be resolved by the model. The unresolved processes, so-called subgrid-scale processes, typically cannot be omitted since due to the nonlinear nature of the governing equations, effects of these processes can have an influence on the resolved grid-scale processes. Approximately including unresolved processes in the resolved dynamics of a numerical model is referred to as parameterization.

Standard numerical approximations for differential equations typically do not explicitly aim to preserve geometric properties of differential equations at the discrete level. Geometric properties,

such as the invariance of an equation under specific coordinate transformations or conservation laws, are among the most important qualitative features a system of differential equations can possess. Several classical models of physics are built around these fundamental geometric properties. Thus, preserving them at the discrete level is an important task, which is the main aim of geometric numerical integration. Geometric integrators are typically important when long term integrations or statistical properties of differential equations are of concern, since preserving geometric properties such as a Hamiltonian structure or conservation laws is known to guarantee or at least improve numerical stability of a discretization schemes. Therefore, geometric numerical integration plays an eminent role in diverse fields such as astronomy, mathematical biology, molecular dynamics, computer vision and climate modeling.

The purpose of this thesis is to collect several results on my work on discretization and parameterization schemes that preserve Lie symmetries and conservation laws in numerical models related to geophysical fluid dynamics.

We formulate several new methods for preserving Lie symmetries in discretization schemes, such as the invariant adaptive method, the invariant semi-Lagrangian method and the invariant meshless method. The development of these methods was necessary since previously existing methods for constructing invariant discretization schemes were of rather theoretical nature. In particular, several existing invariant descretization schemes develop numerical instabilities almost immediately due to mesh tangling. We illustrate the newly developed methods with several examples, such as Burgers' equation, the Korteweg–de Vries equation and the shallow-water equations.

We also provide several methods for finding invariant and conservative parameterization schemes. Proof-of-concept parameterization schemes are constructed for the barotropic vorticity equation as needed for the simulation of geostrophic turbulence, and the shallow-water equations.

1.2 Structure of the thesis

This thesis is composed out of the following seven papers:

1. A. Bihlo, E.M. Dos Santos Cardoso-Bihlo and R.O. Popovych, 2014. Invariant parameterization and turbulence modeling on the beta-plane. *Phys. D* **269**, 48–62, arXiv:1112.1917.

2. A. Bihlo and G. Bluman, 2013. Conservative parameterization schemes. *J. Math. Phys.* **54**, 083101 (24 pp), arXiv:1209.4279.

3. A. Bihlo and R.O. Popovych, 2012. Invariant discretization schemes for the shallow-water equations. *SIAM J. Sci. Comput.* **34** (6), B810–B839, arXiv:1201.0498.

4. A. Bihlo and J.-C. Nave, 2013. Invariant discretization schemes using evolution-projection techniques. *SIGMA* **9**, 052, 23 pp, arXiv:1209.5028.

5. A. Bihlo and J.-C. Nave, 2014. Convecting reference frames and invariant numerical models. *J. Comput. Phys.* **271**, 656–663, arXiv:1301.5955.

6. A. Bihlo, X. Coiteux-Roy and P. Winternitz, 2015. The Korteweg–de Vries equation and its symmetry-preserving discretization. *J. Phys. A* **48**, 055201 (25 pp), arXiv:1409.4340.

7. A. Bihlo, 2013. Invariant meshless discretization schemes. *J. Phys. A* **46** (6), 062001 (12 pp), arXiv:1210.2762.

Thematically, the first two paper are devoted to the problem of geometric parameterization, the remaining five papers are devoted to the problem of geometric discretization.

Specifically, in the **first paper** (Chapter 2) we propose the use of equivariant moving frames to invariantize existing parameterization schemes. Here we start with a given non-invariant parameterization and map it to an invariant parameterization using a suitable moving frame. The procedure is illustrated by invariantizing hyperdiffusion models for the barotropic vorticity equation on the beta-plane.

In the **second paper** (Chapter 3) we lay out a theory for the construction of conservative parameterization schemes. Here we identify the problem of conservative parameterization as a classification problem for conservation laws. As an example, we propose conservative parameterization schemes for the shallow-water equations. A brief discussion on combined invariant and conservative parameterization schemes is included as well.

The **third paper** (Chapter 4) discusses the construction of invariant discretization schemes for both the (1+1)-dimensional shallow-water equations and the (1+2)-dimensional shallow-water equations. The latter provides one of the first example of an invariant discretization scheme for a system of multi-dimensional evolution equations.

In the **fourth paper** (Chapter 5) we develop a new method for constructing invariant discretization schemes that is essentially an invariant version of a semi-Lagrangian discretization method, consisting of an invariant evolution step and an invariant projection step. The new method is illustrated for the (1+1)-dimensional linear heat equation.

The **fifth paper** (Chapter 6) continues the ideas of Chapter 5 and extends them to the example of Burgers' equation. Here we also present some numerical evidence showing that fully invariant discretization schemes for Burgers' equation outperform previously proposed semi-invariant schemes.

In the **sixth paper** (Chapter 7) we apply various methods for finding invariant discretization schemes for the Korteweg–de Vries (KdV) equation. This problem was timely since the invariant discretization schemes previously proposed for the KdV equation in the literature would almost immediately lead to mesh tangling and hence are not suitable for practical numerical integrations.

In the **seventh paper** (Chapter 8) we propose a new method for finding invariant discretization schemes, which we called the invariant meshless method. As an illustration to the general idea, we construct an invariant numerical scheme for nonlinear diffusion equation and show that this scheme outperforms a similar non-invariant scheme.

1.3 Further work

The results presented in this thesis represents only a cross-section of the research I carried out since finishing my Ph.D. thesis in 2010. In particular, the large body of work on the development of various techniques for the group classification of differential equations and conservation laws has been omitted from this thesis. Likewise, my work on the use of domain decomposition for adaptive grid generation, the development of numerical schemes for the shallow-water equations as used in tsunami modeling, and the development of conservative discretization schemes is not part of this manuscript. The decision for omission of this work was made to present only one coherent stream of my activities—the use of geometric properties in the development of numerical methods for the equations of fluid dynamics—in its entirety.

Part I

Structure-preserving parameterization

Chapter 2

Invariant parameterization and turbulence modeling on the beta-plane

Alexander Bihlo[†1], Elsa Dos Santos Cardoso-Bihlo[‡2] and Roman O. Popovych[‡§3]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*

[‡] *Wolfgang Pauli Institute, Oskar-Morgenstern-Platz 1, A-1090 Vienna, Austria*

[§] *Institute of Mathematics of NAS of Ukraine, 3 Tereshchenkivska Str., 01601 Kyiv, Ukraine*

E-mail: [1]bihlo@crm.umontreal.ca, [2]elsa.cardoso@univie.ac.at, [3]rop@imath.kiev.ua

Invariant parameterization schemes for the eddy-vorticity flux in the barotropic vorticity equation on the beta-plane are constructed and then applied to turbulence modeling. This construction is realized by the exhaustive description of differential invariants for the maximal Lie invariance pseudogroup of this equation using the method of moving frames, which includes finding functional bases of differential invariants of arbitrary order, a minimal generating set of differential invariants and a basis of operators of invariant differentiation in an explicit form. Special attention is paid to the problem of two-dimensional turbulence on the beta-plane. It is shown that classical hyperdiffusion as used to initiate the energy–enstrophy cascades violates the symmetries of the vorticity equation. Invariant but nonlinear hyperdiffusion-like terms of new types are introduced and then used in the course of numerically integrating the vorticity equation and carrying out freely decaying turbulence tests. It is found that the invariant hyperdiffusion scheme is closely reproducing the theoretically predicted k^{-1} shape of enstrophy spectrum in the enstrophy inertial range. By presenting conservative invariant hyperdiffusion terms, we also demonstrate that the concepts of invariant and conservative parameterizations are consistent.

2.1 Introduction

As atmospheric and oceanic numerical models get increasingly complex, it becomes more and more challenging to propose valuable conceptual paradigms for those processes that the model

is still not able to capture owing to its limited spatial and temporal resolution. This problem is common to all numerical models irrespectively of their eventual degree of sophistication [34, 35]. In the beginning of numerical modeling in geophysical fluid dynamics, it was often the lack of computer power that dictated which processes had to be parameterized, even with a concise understanding of these processes. As computers became more capable, the problem of parameterization shifted to processes occurring on rather fine scales where it can be difficult to retrieve accurate experimental data. Accordingly, for various processes that should be taken into account in order to improve the forecast range of a numerical model, there is still no satisfactory general understanding. This naturally makes it difficult to set up valuable parameterization schemes, which for this reason is usually an elaborate task.

On the other hand, processes that occur in geophysical fluid dynamics and that can be described using differential equations also might have certain structural or geometrical properties. Such properties can be conservation of mass or energy or other fundamental conservation laws. Real-world processes are generally also invariant under specific transformation groups, as e.g. the Galilean group. This is why one can ask the question whether it is reasonable to construct parameterization schemes for processes possessing certain structural features in a manner such that these features are preserved in the closed model. In this way, even if a model is not able to explicitly resolve processes, loosely speaking, it takes into account some of their significant properties. This study was initiated in [20] for the problem of finding invariant turbulence closure schemes for the filtered Navier–Stokes equations. In the present paper we aim to give a further instance for invariant parameterization schemes by constructing closure ansatzes that retain certain Lie point symmetries of the barotropic vorticity equation on the beta-plane.

This possible stream of constructing geometrically motivated parameterization schemes in some sense parallels the present general trend in numerical modeling to design specially adapted discretizations of differential equations that capture a range of their qualitative or global features, such as conservation laws, a Hamiltonian structure or symmetry properties. Especially the possibility of constructing discretization schemes that have the same symmetries and/or conservation laws as the original differential equations they are a model of, as proposed and discussed e.g. in [6, 11, 12, 21, 22, 45, 50, 52], is of immediate relevance to the present work. This is because, strictly speaking, a discretization of a system of differential equation is in practice not enough to set up a valuable numerical model. There always has to be a model for the unresolved parts of the dynamics. (Neglecting them is in general not a good idea as for nonlinear differential equations these parts will, sooner or later, spoil the numerical integration.) Then, if one aims to construct an invariant discrete counterpart of some relevant physical model, care should also be taken about the invariance characteristics of the processes that are not explicitly resolved. This is where the method of finding invariant parameterization schemes comes into play. The combination of invariant discretization schemes for the resolved part of the model with invariant parameterization schemes for the unresolved parts yields a completely invariant numerical description of a given system of differential equations. Such a fully invariant model might be closer to a true geometric numerical integration scheme than solely a symmetry preserving discretization without any closure for the subgrid-scale terms or with some non-invariant closure.

Perhaps the most relevant usage of the barotropic vorticity equation is related to two-dimensional turbulence. Turbulence on the beta-plane (or, more general, on the rotating sphere)

is peculiar in that it allows for the combination of turbulent and wave-like effects. It is believed to explain the emergence of strong jets and band-like structure on giant planets in our solar system and is therefore the subject of intensive investigation, see e.g. [24, 32, 46, 49, 57] and references therein. In the present paper we focus on freely decaying turbulence on the beta-plane by using invariant hyperdiffusion terms to initiate the energy–enstrophy cascades. These cascades are likely responsible for the emergence of coherent, stable structures (vortices) that are ubiquitous in large-scale geophysical fluid dynamics. Note that the possibility of a development of coherent structures causing the classical inverse cascade to break down is also discussed in the literature. Energy can then be transferred between different scales without a non-linear cascade [47].

The outline of the paper is as follows. In the subsequent Section 2.2, we discuss and slightly extend the concept of invariant parameterization schemes as introduced in [20] and [26]. Special attention will be paid to methods related to invariant parameterization schemes and inverse group classification. In Section 2.3 we present the maximal Lie invariance algebra $\mathfrak{g}_1$ and the maximal Lie invariance pseudogroup of the barotropic vorticity equation on the beta-plane. The computation of the algebra $\mathfrak{g}_1$ is briefly described in Appendix 2.A. A concise description of the general method for computing differential invariants of Lie (pseudo)groups using the method of moving frames is given in Section 2.4. In Section 2.5 the algebra of differential invariants is determined for the maximal Lie invariance pseudogroup of the vorticity equation. The related computation can be found in Appendix 2.B. Two examples for invariant parameterization schemes constructed out of existing schemes using the invariantization process are presented in Section 2.6. Section 2.7 is devoted to the application of differential invariants in turbulence on the beta-plane. In particular, invariant hyperdiffusion schemes are introduced. The vorticity equation on the beta-plane is integrated numerically using both invariant and non-invariant hyperdiffusion and the corresponding enstrophy spectra are obtained. In Section 2.8 we discuss the possibility of deriving invariant parameterization schemes that also respect conservation laws. As an example, an invariant diffusion term is constructed that preserves the entire maximal Lie invariance pseudogroup of the vorticity equation and also preserves conservation of energy, circulation and momentum. The results are summarized and further discussed in the final Section 4.6, in which we also indicate possible future research directions in the field of invariant parameterization.

2.2 Invariant parameterization schemes

The problem of finding parameterization or closure schemes for subgrid-scale terms in averaged differential equations that admit Lie symmetries of the original (unaveraged) differential equation was first raised in [20], see also [34, 44]. Recently, we put this idea into the framework of group classification [26], by showing that any problem of constructing invariant parameterization schemes amounts in solving a (possibly complicated) group classification problem.

As for the classical group classification, there are two principal ways to construct parameterization schemes, the direct and the inverse one [26]. In the direct approach, one replaces the terms to be parameterized with arbitrary functions depending on the mean variables and derivatives thereof. This is in the line with the general definition of all physical parameterization schemes, which are concerned to express the unknown subgrid-scale terms using the information included in the grid-scale (mean) quantities. The form of dependency of the arbitrary functions on the mean variables is guided by physical intuition. It determines the properties of all the

families of invariant parameterization schemes that can be derived (e.g. the highest order of derivatives that can arise). Once the general form of the arbitrary function is chosen, one is left with a possibly rather general class of differential equations, which is amenable with tools from usual group classification, see e.g. [5, 14, 27]. This in particular will lead to a list of families of mutually inequivalent parameterization schemes that admit different Lie invariance algebras. One can then select those families that preserve the most essential symmetry features of the process to be parameterized. The final step is to suitably narrow the selected families by including other desirable physical properties into the invariant parameterization scheme.

In the present paper, however, we will be solely concerned with the inverse approach, which is why we will discuss it in a more extended manner. The inverse approach rests on the fact that any system of differential equations can be rewritten in terms of differential invariants of its maximal Lie invariance group, provided that the prolongation of the group to the corresponding jet space acts semi-regularly [42]. This property can be used in the course of the parameterization problem in the following way: Suppose that we are given a Lie group G regarded as important to be preserved for valuable parameterization schemes as a Lie symmetry group. Computing a basis of differential invariants of G along with a complete set of its independent operators of invariant differentiation, see e.g. [9, 10, 18, 30], serves to exhaustively describe the entire algebra of differential invariants of G. As any combination of these differential invariants will necessarily be invariant with respect to G, assembling them together to a parameterization will immediately lead to a closure scheme admitting G as a Lie symmetry group.

The key question hence lies in the correct selection of a suitable symmetry group. The initial point for the selection is given by symmetry properties of the model to be parameterized. In the course of the parameterization one can intend to preserve the whole Lie symmetry group of the initial model or its proper subgroups. The choice for an invariance group for parameterization obviously should not solely be justified using mathematical arguments. Sometimes, it can be motivated from obvious physical reasons. If the process to be parameterized can be described within the framework of classical mechanics then any reasonable parameterization for that process should be invariant under the Galilean group. Moreover, for turbulence closure schemes, scale invariance might be of particular importance. For processes that can be described within the framework of a variational principle and respect certain conservation laws, it might be reasonable that the parameterization scheme to be developed respects the associated Noether symmetries. See [6, Chapter 6] for similar studies of discretization schemes within the variational framework.

There are several processes in fluid mechanics that are intimately linked to the presence of certain boundary conditions (e.g., turbulence near walls, boundary layer convection, etc.). For such processes the inclusion of the particular boundaries is an integral part in the formulation of a parameterization scheme. At first glance, to find invariant parameterization schemes for such processes it is inevitable to single out those subgroups of the maximal Lie invariance group G of the system $\mathcal{L}$ of differential equations describing the process of interest that are compatible with a particular boundary value problem. The main complication with this approach is that most of boundary value problems admit no symmetries, see e.g. [4]. At the same time, it is more natural to assume that symmetries of $\mathcal{L}$ act as equivalence transformations on a joint class of physically relevant boundary value problems for $\mathcal{L}$, i.e., these transformations send a particular boundary value problem to another problem from the same class [4]. Even the basic physical symmetries including shifts in space and time, rotations, scalings, Galilean boosts or

Lorentz transformations, which are related to fundamental properties of the space and the time (homogeneity, isotropy, similarity, Galilean or special relativity principle, respectively), usually act on boundary value problems in much the same way as equivalence transformations. This is why it is the generation of a group of well-defined equivalence transformations on a properly chosen class of boundary value problems that can serve as a criterion for selecting a subgroup of G to be taken into account in the course of invariant parameterization of $\mathcal{L}$.

Employing techniques of inverse group classification does not automatically lead to ready-to-use parameterizations, but it gives a frame in which parameterizations can be defined without the violation of basic invariance properties. Examples of the violation have been reported in the recent literature. See, e.g., [20] for a discussion about the Smagorinsky model in the filtered Navier–Stokes equations violating scale invariance and [26] for a note on the Kuo convection schemes that describes a Galilean invariant process in a non-invariant fashion. The construction of parameterization schemes that fail to preserve essential symmetries can be easily avoided by applying the above methods of inverse group classification. This may help to restrict the large number of possible closure schemes using geometrical reasoning and thereby may assist in finding a proper description of unresolved subgrid-scale processes.

There is yet a second possibility to construct invariant parameterization schemes using the inverse group classification approach, which has not been reported in [26]. It rests on the construction of *moving frames* for the Lie group G with respect to which parameterizations under study should be invariant. It is a general feature of a moving frame that it allows constructing of invariant counterparts of differential functions. This property enables the construction of an invariant parameterization scheme out of a non-invariant one. It is simply necessary to apply the moving frame corresponding to the selected Lie group G to the specific closed differential equation. More precisely, consider a system $\mathcal{L}$ of differential equations

$$L^l(x, u_{(n)}) = 0, \quad l = 1, \dots, m.$$

The dependent variables u can be represented according to $u = \bar{u} + u'$, where $\bar{u}$ is the averaged or filtered part of the dynamics (i.e. the resolved or grid-scale part) and u' denotes the departure of u from the mean or filtered part $\bar{u}$ (i.e. the subgrid-scale part). Numerical models in geophysical fluid dynamics are formulated as equations for the resolved part, which are obtained from $L^l = 0$ by averaging or filtering, leading to

$$\tilde{L}^l(x, \bar{u}_{(n)}, w) = 0, \quad l = 1, \dots, m,$$

where $\tilde{L}^l$ are smooth differential functions of their arguments. The particular form of $\tilde{L}^l$ depends on the actual averaging rule chosen and the form of the initial system $\mathcal{L}$. The unknown subgrid-scale terms that arise in the course of averaging (e.g. by using the Reynolds averaging rule for products, $\overline{ab} = \bar{a}\bar{b} + \overline{a'b'}$) are collected in the tuple w. These terms have to be parameterized in order to close the above system of averaged differential equations. A local parameterization scheme establishes a particular functional relation

$$w = \theta(x, \bar{u}_{(r)})$$

between the subgrid-scale and grid-scale quantities. Let there be given a moving frame $\rho^{(j)}$ of order $j = \max(r, n)$ for the selected Lie group G, see Section 2.4. Any particular parameterization scheme can then be invariantized via replacing $\tilde{L}^l$ and θ by their invariantized counterparts,

$$\iota(\tilde{L}^l(x, \bar{u}_{(n)}, w)) = \tilde{L}^l(\rho^{(j)} \cdot x, \rho^{(j)} \cdot \bar{u}_{(n)}, w) \quad \text{and} \quad \iota(\theta(x, \bar{u}_{(r)})) = \theta(\rho^{(j)} \cdot x, \rho^{(j)} \cdot \bar{u}_{(r)}).$$

Example 1. It is instructive to illustrate this idea with a simple example. Let us consider the famous Korteweg–de Vries (KdV) equation,

$$u_t + uu_x + u_{xxx} = 0.$$

Its maximal Lie invariance group G_{KdV} is four-dimensional and the most general transformation leaving the KdV equation invariant is

$$(T, X, U) = (e^{3\varepsilon_4}(t + \varepsilon_1), e^{\varepsilon_4}(x + \varepsilon_2 + \varepsilon_1\varepsilon_3 + \varepsilon_3 t), e^{-2\varepsilon_4}(u + \varepsilon_3)), \tag{2.1}$$

where $\varepsilon_1, \ldots \varepsilon_4$ are arbitrary constants. Let us now apply the classical Reynolds averaging to the KdV equation. This yields

$$\bar{u}_t + \bar{u}\bar{u}_x + \bar{u}_{xxx} = -\frac{1}{2}(\overline{u'^2})_x,$$

where the right-hand side is the term we seek closure for. A simple closure ansatz is the down-gradient parameterization, i.e. we close the above equation by setting $\overline{u'^2}/2 = -\kappa\bar{u}_x$, where for the sake of simplicity we use $\kappa = \mathrm{const}$. This yields the closed KdV equation

$$\bar{u}_t + \bar{u}\bar{u}_x + \bar{u}_{xxx} = \kappa\bar{u}_{xx}. \tag{2.2}$$

However, it is easily verified that this equation is not invariant under the transformation (2.1). Namely, the scale invariance is lost, i.e. the closed KdV equation is invariant only under the three-parameter group of transformations associated with the group parameters ε_1, ε_2 and ε_3. To restore scale invariance, we can invariantize the closed KdV equation (2.2) using the moving frame associated with the group G_{KdV}.

Moving frames for the group G_{KdV} were constructed in [9, 18]. It is convenient to invariantize Eq. (2.2) using the moving frame with

$$\varepsilon_1 = -t, \quad \varepsilon_2 = -x, \quad \varepsilon_3 = -\bar{u}, \quad \varepsilon_4 = \frac{1}{3}\ln\bar{u}_x.$$

This is done by firstly applying the transformations (2.1) to (2.2) which yields

$$\bar{u}_t + \bar{u}\bar{u}_x + \bar{u}_{xxx} = \kappa e^{\varepsilon_4}\bar{u}_{xx},$$

showing explicitly that this equation fails to be scale invariant. The invariantization is completed upon substituting the moving frame for ε_4 giving

$$\bar{u}_t + \bar{u}\bar{u}_x + \bar{u}_{xxx} = \kappa\sqrt[3]{\bar{u}_x}\bar{u}_{xx}.$$

It is readily checked that this closed equation is invariant under the same symmetry group G_{KdV} as is the original KdV equation. The price for restoring scale invariance of the closed KdV equation invoking the simple down-gradient parameterization is that the closure scheme becomes nonlinear. We will observe the same effect when invariantizing linear hyperdiffusion models for the vorticity equation on the beta-plane, which will be shown in detail below.

2.3 Lie symmetries of the vorticity equation

The barotropic vorticity equation on the beta-plane is a simple but still genuine meteorological model. It has the form

$$\zeta_t + \psi_x\zeta_y - \psi_y\zeta_x + \beta\psi_x = 0, \quad \text{or} \quad \zeta^{\mathrm{a}}_t + J(\psi, \zeta^{\mathrm{a}}) = 0. \tag{2.3}$$

Here $J(a,b) := a_x b_y - a_y b_x$, $\psi = \psi(t,x,y)$ is the stream function, $\zeta = \psi_{xx} + \psi_{yy}$ is the vorticity and $\zeta^{\mathrm{a}} = \zeta + \mathsf{f} = \zeta + \mathsf{f}_0 + \beta y$ is the absolute vorticity under the β-plane approximation $\mathsf{f} = \mathsf{f}_0 + \beta y$ of the Coriolis parameter f, β is a nonzero constant parameter (the differential rotation). The constant f_0 is dynamically inessential and can be neglected.

The maximal Lie invariance algebra $\mathfrak{g}_1$ of Eq. (2.3) is spanned by the vector fields

$$\mathcal{D} = t\partial_t - x\partial_x - y\partial_y - 3\psi\partial_\psi, \quad \partial_t, \quad \partial_y, \quad \mathcal{X}(\tilde f) = \tilde f(t)\partial_x - \tilde f_t(t) y\partial_\psi \quad \mathcal{Z}(\tilde g) = \tilde g(t)\partial_\psi,$$

where the parameters $\tilde f$ and $\tilde g$ run through the set of smooth functions of t [5, 31]. More details on how the above vector fields are obtained can be found in Appendix 2.A. The vorticity equation (2.3) also admits two independent discrete symmetries, which alternate signs of the pairs (t,x) and (y,ψ), see [9] for more details. Such discrete symmetries will not be taken into account in the course of construction of differential invariants. Any nonzero value of β can be gauged to one by a scaling transformation.

The one-parameter Lie (pseudo)groups generated by the above vector fields read

$$\begin{aligned}
&\Gamma_{\varepsilon_1}\colon && (t,x,y,\psi) \mapsto (e^{\varepsilon_1}t, e^{-\varepsilon_1}x, e^{-\varepsilon_1}y, e^{-3\varepsilon_1}\psi)\\
&\Gamma_{\varepsilon_2}\colon && (t,x,y,\psi) \mapsto (t+\varepsilon_2, x, y, \psi)\\
&\Gamma_{\varepsilon_3}\colon && (t,x,y,\psi) \mapsto (t, x, y+\varepsilon_3, \psi)\\
&\Gamma_{f}\colon && (t,x,y,\psi) \mapsto (t, x+f(t), y, \psi - f_t(t)y)\\
&\Gamma_{g}\colon && (t,x,y,\psi) \mapsto (t, x, y, \psi + g(t)),
\end{aligned}$$

where $\varepsilon_i \in \mathbb{R}$ and $f(t) := \varepsilon_4 \tilde f(t)$ and $g(t) := \varepsilon_5 \tilde g(t)$. Accordingly, the admitted Lie symmetries of the barotropic vorticity equation on the beta-plane are scalings, time translations, translations in y-direction, generalized Galilean boosts in the x-direction and gaugings of the stream function with smooth time-dependent summands.

We will compose transformations from these one-parameter Lie (pseudo)groups in the following way $\Gamma = \Gamma_{\varepsilon_1} \circ \Gamma_{\varepsilon_2} \circ \Gamma_{\varepsilon_3} \circ \Gamma_f \circ \Gamma_g$ to a transformation Γ from the maximal Lie symmetry pseudogroup G_1 of the vorticity equation (2.3). Any transformation of G_1 then has the form

$$(T,X,Y,\Psi) = \big(e^{\varepsilon_1}(t+\varepsilon_2),\, e^{-\varepsilon_1}(x+f(t)),\, e^{-\varepsilon_1}(y+\varepsilon_3),\, e^{-3\varepsilon_1}(\psi + g(t) - f_t(t)y)\big). \tag{2.4}$$

In what follows, we set $h(t,y) = g(t) - f_t(t)y$ for convenience and use the substitution $h_y = -f_t$, whenever h_y occurs.

Note that the maximal Lie invariance algebra $\mathfrak{g}_0$ of the usual vorticity equation, which is also called the barotropic vorticity equation on the f-plane and corresponds to the value $\beta = 0$, is much wider than the algebra $\mathfrak{g}_1$ and contains $\mathfrak{g}_1$ as a proper subalgebra [1, 2]. The algebra $\mathfrak{g}_0$ is spanned by the vector fields from $\mathfrak{g}_1$ jointly with the vector fields

$$t\partial_t - \psi\partial_\psi, \quad -y\partial_x + x\partial_y, \quad -ty\partial_x + tx\partial_y + \tfrac12(x^2+y^2)\partial_\psi, \quad \tilde h(t)\partial_y + \tilde h_t(t)x\partial_\psi,$$

where the parameter $\tilde h$ runs through the set of smooth functions of t. This means that in addition to the transformations from G_1 the maximal Lie symmetry pseudogroup G_0 of the usual vorticity equation also contains one more family of scalings, usual rotations in the (x,y)-plane, rotations depending on t with constant angle velocities and generalized Galilean boosts in the y-direction.

Remark 2.1. In order to set up a numerical model, a decision has to be taken about which boundary conditions should be implemented. The numbers of symmetries admitted by a differential equation is almost always reduced for an associated boundary value problem. The most

immediate boundary conditions in the atmospheric sciences are periodic ones. However, a periodic domain implies a fixed domain size and therefore breaks the scale invariance of Eq. (2.3). On the other hand, scale invariance is an equivalence transformation of the class of *all* periodic boundary value problems of the vorticity equation, see also Section 4.1 in [4]. A more serious problem is that the periodicity in y-direction is not natural for the beta-plane from the physical point of view. At the same time, using a channel model (rigid walls in the North and in the South of the domain) breaks also the translational invariance in y-direction thereby reducing the admitted Lie symmetry group even stronger than in the presence of doubly periodic boundary conditions (though, in contrast to usual hyperdiffusion, it would not be necessary to define an additional boundary condition for the invariant hyperdiffusion as by definition $\psi_x = 0$ at the walls of the channel and the diffusion term therefore vanishes there). This is why we will use doubly periodic boundary conditions although $\beta \neq 0$ here. Despite this slight inconsistency, doubly periodic boundary conditions are used quite extensively in studying turbulence properties on the beta-plane [32, 46, 57].

The above form (2.3) of the vorticity equation is not particularly useful for a numerical evaluation. The reason is, of course, that any numerical model can be run only at a finite resolution, which requires a suitably chosen averaging or filtering of Eq. (2.3). As from the point of view of invariant parameterization schemes the precise type of averaging is only of secondary importance, we will employ a classical Reynolds averaging to Eq. (2.3) in the paper. This leads to the averaged vorticity equation

$$\bar{\zeta}_t + \bar{\psi}_x\bar{\zeta}_y - \bar{\psi}_y\bar{\zeta}_x + \beta\bar{\psi}_x = \overline{(\zeta'\psi'_y)}_x - \overline{(\zeta'\psi'_x)}_y, \tag{2.5}$$

where the right-hand side of this equation denotes the eddy-vorticity flux, which we aim to parameterize subsequently. For the sake of notational simplicity, we will omit bars over the mean quantities from now on.

Slightly more generally, the vorticity equation (2.3) can be augmented with external forcings F and dissipative terms D yielding a general expression of the form

$$\zeta_t + \psi_x\zeta_y - \psi_y\zeta_x + \beta\psi_x = F + D. \tag{2.6}$$

A further question we aim to address is whether symmetries might be helpful in deriving invariant expressions for F and D. As by definition F denotes *external* forcing terms, it is not immediately clear why symmetries of the vorticity equation should place restrictions on the form of F. However, as we shall show, symmetries are valuable in finding invariant diffusion terms D that can be used in the course of turbulence modeling. For the sake of simplicity we therefore will use Eq. (2.6) for the case of $F = 0$ and $D \neq 0$, i.e. we assume that no external forcing acts on the system to which a damping is attached. Physically, the presence of F and D can be interpreted as symmetry breaking in the vorticity equation (2.3). Which symmetries are to be broken and which are to be preserved can be controlled upon expressing the term D via differential invariants derived in Section 2.5. This is a comprehensive problem and not all of the cases arising might be interesting from the physical point of view. We therefore restrict ourselves on the case where D or the eddy vorticity flux in Eq. (2.5) can be represented in such a manner that the resulting equation admits all the transformations from the maximal Lie invariance pseudogroup (2.4). This is the approach proposed in [20] and it appears to be suitable for the beta-plane equation.

2.4 Algorithm for the construction of differential invariants

Given a transformation pseudogroup G in the space of p independent variables $x = (x^1, \dots, x^p)$ and q dependent variables $u = (u^1, \dots, u^q)$, the exhaustive description of its differential invariants is reduced to either the construction of a functional basis of differential invariants of any fixed order or finding a complete set of independent operators of invariant differentiation and a minimal set of differential invariants that generate all differential invariants through invariant differentiation and functional combination [30, Section 24]. Within the framework of the method of moving frames the solution of this problem is split into two parts [9]. It is convenient to compute normalized differential invariants and operators of invariant differentiation using the explicit expressions for transformations from G. The corresponding computation consists of two procedures, *normalization* and *invariantization*. At the same time, the derivation of syzygies (i.e., relations involving operators of invariant differentiation) between normalized differential invariants is mostly based on the determining equations of G, and an important tool for this is given by *recurrence formulas*. In this section we briefly describe related basic notions and results, paying the main attention to the computational realization of algorithms in fixed local coordinates. See [9, 9, 10, 18, 19, 39] for detail and rigorous presentations.

In what follows the index j runs from 1 to p, the index a runs from 1 to q. We use two kinds of integer tuples for the indexing of objects. One of these kinds is given by the usual multi-indices of the form $\alpha = (\alpha_1, \dots, \alpha_p)$, where $\alpha_j \in \mathbb{N}_0 = \mathbb{N} \cup \{0\}$ and $|\alpha| = \alpha_1 + \dots + \alpha_p$. By δ_j we denote the p-index whose jth entry equals 1 and whose other entries are zero. Thus, both the derivative $\partial^{|\alpha|} u^a / (\partial x^1)^{\alpha_1} \cdots (\partial x^p)^{\alpha_p}$ and the associated variable of the jet space $\mathrm{J}^\infty(x|u)$ are denoted by u^a_α, $\mathrm{D}^\alpha = \mathrm{D}_1^{\alpha_1} \cdots \mathrm{D}_p^{\alpha_p}$, etc. Here $\mathrm{D}_j = \partial_{x^j} + \sum_{\alpha,a} u^a_{\alpha+\delta_j} \partial_{u^a_\alpha}$ is the operator of total differentiation with respect to the variable x^j. The other kind of index tuples is presented by $J = (j_1, \dots, j_\kappa)$, where $1 \leqslant j_k \leqslant p$, $k = 1, \dots, \kappa$, $\kappa \in \mathbb{N}_0$. Such index tuples are used for the indexing of compositions of operators of invariant differentiation, which do not commute. Namely, we write $\mathrm{D}^{\mathrm{i}}_J$ for $\mathrm{D}^{\mathrm{i}}_{j_1} \cdots \mathrm{D}^{\mathrm{i}}_{j_\kappa}$. The symbol $\mathrm{d_h}$ denotes the horizontal differential, $\mathrm{d_h} F = \sum_{j=1}^p (\mathrm{D}_j F) \mathrm{d}x^j$ for a differential function $F = F[u]$, i.e. a function of x^j and u^a_α.

The normalization procedure for the pseudogroup G consists of three steps:

1. Choose a parameterization (local coordinates) of G and find explicit formulas for the prolonged action of G in terms of the jet variables.

2. Choose a subset of the transformed jet variables and equate the expressions for them to chosen constants.

3. Solve the obtained system of normalization equations as a system of algebraic equations with respect to the parameters of the pseudogroup G including the derivatives of the functional parameters.

The second step is nothing but a choice of an appropriate (coordinate) cross-section of the G-orbits. This should be implemented in a way ensuring that the system from the third step will be well defined and solvable.

The normalization procedure results in the construction of a moving frame ρ for the pseudogroup G, which is, roughly speaking, an equivariant map from the jet space to G. Once the moving frame is constructed it can be used to map any object $\chi(x, u_{(n)})$ defined on an open subset of the jet space (a differential function, a differential operator or a differential form) to its invariant counterpart, $\iota(\chi(x, u_{(n)})) = \chi(\rho^{(n)}(x, u_{(n)}) \cdot (x, u_{(n)}))$. To carry out this in practice,

one should replace all occurrences of the pseudogroup parameters in the transformed version of the object by their expressions obtained with the normalization procedure.

Thus, the invariantization of the coordinate functions x^j and u^a_α of the jet space yields the so-called *normalized differential invariants* $H^j = \iota(x^j)$ and $I^a_\alpha = \iota(u^a_\alpha)$. In fact, the invariantized coordinate functions whose transformed counterparts were used to set up the normalization equations are equal to the respective constants chosen in the course of normalization and hence these objects are called *phantom differential invariants*. Non-phantom normalized differential invariants are functionally independent and any differential invariant can be represented as a function of normalized differential invariants. Invariantization of the *operators of total differentiation*, D_j, gives the operators of invariant differentiation, $\mathrm{D}^{\mathrm{i}}_j$, which upon acting on differential invariants produce other differential invariants. Note that the domain of the jet space, where invariantized objects are well defined, depends on what cross-section is chosen.

In order to determine the algebra of differential invariants the normalized differential invariants and the operators of invariant differentiation play a key role. It has been proved [30] that for any Lie (pseudo)group the algebra of differential invariants can be completely described upon finding a *finite* generating set of differential invariants. As stated above, all the other differential invariants are then a suitable combination of the basis differential invariants or their invariant derivatives. The hardest part in describing the algebra of differential invariants is usually to find a *minimal* generating set of these invariants. Proving the minimality of a given basis usually involves the computation of the *syzygies* among the differential invariants, meaning functional relations among the differentiated differential invariants $\mathrm{D}^{\mathrm{i}}_J I^a_\alpha$, $S(\dots, \mathrm{D}^{\mathrm{i}}_J I^a_\alpha, \dots) = 0$.

In general, the normalized differential invariants are derived from invariantization of the derivatives of the dependent variables, whereas the differentiated differential invariants are obtained by acting on normalized differential invariants of lower order with the operators of invariant differentiation. The central point is that the operations of invariant differentiation and invariantization of a differential function in general do not commute. Roughly speaking, the failure of commutation of these two operations is quantified by the so-called recurrence relations

$$\mathrm{d_h} H^j = \omega^j + \hat{\xi}^j, \quad \mathrm{d_h} I^a_\alpha = \sum_{j=1}^{p} I^a_{\alpha+\delta_j} \omega^j + \hat{\varphi}^{a,\alpha}, \tag{2.7}$$

where $\omega^j = \iota(\mathrm{d}x^j)$ [9, 19]. The forms $\hat{\xi}^j = \iota(\xi^j)$ and $\hat{\varphi}^\alpha_a = \iota(\varphi^\alpha_a)$ are the invariantizations of the coefficients of the general prolonged infinitesimal generator

$$Q_\infty = \sum_{j=1}^{p} \xi^j \partial_{x^j} + \sum_{\alpha \geqslant 0} \sum_{a=1}^{q} \varphi^{a,\alpha} \partial_{u^a_\alpha}, \quad \varphi^{a,\alpha} = \mathrm{D}^\alpha \left(\varphi^a - \sum_{j=1}^{p} \xi^j u^a_{\delta_j} \right) + \sum_{j=1}^{p} \xi^j u^a_{\alpha+\delta_j},$$

of G. More rigorously, here ξ^j and u^a_α are interpreted as coordinate functions on the space of prolonged infinitesimal generators of G, i.e., first-order differential forms in the jet space. Hence their invariantizations should also be forms, which are called *invariantized Maurer–Cartan forms*.

The left-hand sides of the relations (2.7) are zero for phantom differential invariants. If the cross-section is chosen in a proper way, the recurrence relations for the phantom invariants can be solved for the independent invariantized Maurer–Cartan forms, which in turn can be plugged into the relations for the non-phantom differential invariants. Collecting coefficients of ω^j then yields a closed description of the relation between normalized and differentiated differential invariants, which in turn might enable the determination of a basis of differential invariants. For this latter task, specialized methods from computational algebra can be applied [39], which is,

however, not necessary in the present case due to the relatively simple structure of the maximal Lie invariance pseudogroup G_1 of Eq. (2.3).

2.5 Differential invariants for the beta-plane vorticity equation

In order to derive the moving frame for the maximal Lie invariance pseudogroup G_1 of the barotropic vorticity equation on the beta-plane, it is necessary to prolong the group actions to the derivatives of ψ. For this aim, we have to derive expressions for the implicit differentiation operators, D_T, D_X and D_Y. They can be determined as the dual of the lifted horizontal coframe for G_1, which reads

$$\begin{aligned}
&\mathrm{d_h}T = (T_t + \psi_t T_\psi)\mathrm{d}t + (T_x + \psi_x T_\psi)\mathrm{d}x + (T_y + \psi_y T_\psi)\mathrm{d}y = e^{\varepsilon_1}\mathrm{d}t\\
&\mathrm{d_h}X = (X_t + \psi_t X_\psi)\mathrm{d}t + (X_x + \psi_x X_\psi)\mathrm{d}x + (X_y + \psi_y X_\psi)\mathrm{d}y = e^{-\varepsilon_1} f_t \mathrm{d}t + e^{-\varepsilon_1}\mathrm{d}x\\
&\mathrm{d_h}Y = (Y_t + \psi_t Y_\psi)\mathrm{d}t + (Y_x + \psi_x Y_\psi)\mathrm{d}x + (Y_y + \psi_y Y_\psi)\mathrm{d}y = e^{-\varepsilon_1}\mathrm{d}y.
\end{aligned}$$

Therefore, the required implicit differentiation operators are

$$\mathrm{D}_T = e^{-\varepsilon_1}(\mathrm{D}_t - f_t\mathrm{D}_x), \quad \mathrm{D}_X = e^{\varepsilon_1}\mathrm{D}_x, \quad \mathrm{D}_Y = e^{\varepsilon_1}\mathrm{D}_y, \tag{2.8}$$

where D_t, D_x and D_y denote the usual operators of total differentiation with respect to t, x and y, respectively, $\mathrm{D}_t = \partial_t + \sum_\alpha \psi_{\alpha+\delta_1}\partial_{\psi_\alpha}$, $\mathrm{D}_x = \partial_x + \sum_\alpha \psi_{\alpha+\delta_2}\partial_{\psi_\alpha}$ and $\mathrm{D}_y = \partial_y + \sum_\alpha \psi_{\alpha+\delta_3}\partial_{\psi_\alpha}$. Here and in what follows $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ is a multi-index running through $\mathbb{N}_0^3$, $|\alpha| = \alpha_1 + \alpha_2 + \alpha_3$, $\delta_1 = (1,0,0)$, $\delta_2 = (0,1,0)$, $\delta_3 = (0,0,1)$ and the variable $\psi_\alpha = \psi_{\alpha_1\alpha_2\alpha_3}$ of the jet space corresponds to the derivative $\partial^{|\alpha|}\psi/\partial t^{\alpha_1}\partial x^{\alpha_2}\partial y^{\alpha_3}$. We also use the notation $f_{(k)} = \mathrm{d}^k f/\mathrm{d}t^k$ and $h_{(k)} = \partial^k h/\partial t^k$, $k \in \mathbb{N}_0$. The transformed derivatives $\Psi_\alpha = \partial^{|\alpha|}\Psi/\partial T^{\alpha_1}\partial X^{\alpha_2}\partial Y^{\alpha_3}$, $|\alpha| > 0$, are then

$$\begin{aligned}
\Psi_\alpha &= \mathrm{D}_T^{\alpha_1}\mathrm{D}_X^{\alpha_2}\mathrm{D}_Y^{\alpha_3}\Psi = e^{(\alpha_2+\alpha_3-\alpha_1-3)\varepsilon_1}(\mathrm{D}_t - f_t\mathrm{D}_x)^{\alpha_1}\mathrm{D}_x^{\alpha_2}\mathrm{D}_y^{\alpha_3}(\psi + h)\\
&= e^{(\alpha_2+\alpha_3-\alpha_1-3)\varepsilon_1}\left((\mathrm{D}_t - f_t\mathrm{D}_x)^{\alpha_1}\psi_{0\alpha_2\alpha_3} + \left\{\begin{array}{ll} -f_{(\alpha_1+1)}, & \alpha_2 = 0,\ \alpha_3 = 1\\ h_{(\alpha_1)}, & \alpha_2 = \alpha_3 = 0\end{array}\right\}\right).
\end{aligned}$$

We carry out the normalization procedure in the domain of the jet space which is defined by the condition $\psi_x \neq 0$. We choose the conditions for normalization

$$T = X = Y = 0, \quad \Psi_{k00} = \Psi_{k01} = 0, \quad k = 0, 1, \dots, \quad \Psi_{010} = \varepsilon, \tag{2.9}$$

where $\varepsilon = \operatorname{sgn}\psi_x$, which allow us to express all the pseudogroup parameters (including the derivatives of functional pseudogroup parameters) in terms of variables of the jet space:

$$\begin{aligned}
&\varepsilon_1 = \ln\sqrt{|\psi_x|}, \quad \varepsilon_2 = -t, \quad \varepsilon_3 = -y, \quad f = -x,\\
&f_{(k+1)} = (\mathrm{D}_t - \psi_y\mathrm{D}_x)^k\psi_y, \quad h_{(k)} = -(\mathrm{D}_t - \psi_y\mathrm{D}_x)^k\psi, \quad k = 0, 1, \dots.
\end{aligned} \tag{2.10}$$

In other words, the equations (2.10) represent a complete moving frame for the maximal Lie invariance pseudogroup of the vorticity equation. The series of equalities for $f_{(k+1)}$ an $h_{(k)}$ is proved by induction with respect to k using the equations

$$f_{(k+1)} = (\mathrm{D}_t - f_t\mathrm{D}_x)^k\psi_y, \quad h_{(k)} = -(\mathrm{D}_t - f_t\mathrm{D}_x)^k\psi.$$

The nontrivial normalized differential invariants are found via invariantizing the derivatives ψ_α for the values of α for which Ψ_α are not involved in the construction of the above moving frame, i.e., for

$$\alpha \in A = \mathbb{N}_0^3 \setminus \{(k,0,0),\ (k,0,1),\ (0,1,0),\ k \in \mathbb{N}_0\}.$$

In other words, for each $\alpha \in A$ we should substitute the expressions (2.10) for the pseudogroup parameters into the expressions for Ψ_α. (The invariantization of the coordinate functions chosen for the normalization conditions (2.9) are equal to the corresponding normalization constants and are the phantom normalized differential invariants for the moving frame (2.10).) As a result, we obtain the differential invariants

$$I_\alpha = \iota(\psi_\alpha) = |\psi_x|^{(\alpha_2+\alpha_3-\alpha_1-3)/2}(\mathrm{D}_t - \psi_y \mathrm{D}_x)^{\alpha_1}\psi_{0\alpha_2\alpha_3}, \quad \alpha \in A.$$

The order of I_α as a differential function of ψ equals $|\alpha|$. It is also obvious that any finite number of the invariants I_α are functionally independent. This agrees with the general theory of moving frames [9, 9, 19], which also implies a stronger assertion.

Theorem 2.1. *For each $r \geqslant 2$ the functions $I_\alpha = |\psi_x|^{(\alpha_2+\alpha_3-\alpha_1-3)/2}(\mathrm{D}_t-\psi_y\mathrm{D}_x)^{\alpha_1}\psi_{0\alpha_2\alpha_3}$, where $\alpha \in A$ and $|\alpha| \leqslant r$, form a local functional basis of differential invariants of order not greater than r for the maximal Lie invariance pseudogroup G_1 of the barotropic vorticity equation on the beta-plane.*

The description of differential invariants of G_1 given in Theorem 2.1 is sufficient for applications within the framework of invariant parameterization. At the same time, it is interesting and useful to have more information on the structure of the algebra of differential invariants of the pseudogroup G_1 including the operators of invariant differentiation.

Theorem 2.2. *The algebra of differential invariants of the maximal Lie invariance pseudogroup of the barotropic vorticity equation on the beta-plane* (2.3) *is generated, in the domain Ω_1 of the jet space where $\mathrm{D}_x^2(\sqrt{|\psi_x|}\,) \neq 0$, by the single differential invariant $I_{020} = \psi_{xx}/\sqrt{|\psi_x|}$ along with the three operators of invariant differentiation*

$$\mathrm{D}_t^{\mathrm{i}} = \frac{1}{\sqrt{|\psi_x|}}(\mathrm{D}_t - \psi_y\mathrm{D}_x), \quad \mathrm{D}_x^{\mathrm{i}} = \sqrt{|\psi_x|}\mathrm{D}_x, \quad \mathrm{D}_y^{\mathrm{i}} = \sqrt{|\psi_x|}\mathrm{D}_y.$$

All other differential invariants are functions of I_{020} and invariant derivatives thereof. The proof of this theorem is presented in detail in Appendix 2.B.

2.6 Invariantization of parameterization schemes

The Replacement Theorem states that any differential invariant $I(x, u_{(n)})$ of order n can be expressed in terms of the normalized differential invariants via replacing any argument of $I(x, u_{(n)})$ by its respective invariantization, see [10]. In particular, any system of differential equations can be represented using the normalized differential invariants of its associated maximal Lie invariance group. The invariantization of the vorticity equation (2.3) in view of the moving frame (2.10) reads $(I_{120} + I_{102}) + (I_{021} + I_{003}) + \beta = 0$, or, explicitly

$$\frac{\zeta_t - \psi_y\zeta_x}{\psi_x} + \zeta_y + \beta = 0. \tag{2.11}$$

This is the fully invariant representation of the barotropic vorticity equation on the beta-plane.

Differential invariants computed in the previous section can be assembled together to invariant parameterizations of the eddy-vorticity flux in the averaged vorticity equation (2.5). Alternatively, we can invariantize any existing parameterization scheme under the moving frame action (2.10). The following two examples implement this idea.

Example 2. A classical albeit simple parameterization for the eddy-vorticity flux is

$$\mathrm{evf} := \overline{(\zeta'\psi'_y)}_x - \overline{(\zeta'\psi'_x)}_y = \mathrm{D}_x(K\zeta_x) + \mathrm{D}_y(K\zeta_y),$$

where $K = K(x, y)$ might be considered as a spatially dependent function. The most straightforward way to cast this parameterization into the related invariant one is by applying the moving frame (2.10) to the terms on the right-hand side. This yields

$$\begin{aligned}\mathrm{evf}^{\mathrm{i}} &= \mathrm{D}^{\mathrm{i}}_x(K(I_{030} + I_{012})) + \mathrm{D}^{\mathrm{i}}_y(K(I_{021} + I_{003})) = K(I_{040} + 2I_{022} + I_{004})\\ &= K\sqrt{|\psi_x|}(\zeta_{xx} + \zeta_{yy}),\end{aligned}$$

where $\mathrm{evf}^{\mathrm{i}} = \iota(\mathrm{evf})$ and $K = \mathrm{const}$ now as $\iota(x) = \iota(y) = 0$. The invariant representation of the closed barotropic vorticity equation then reads

$$\frac{\zeta_t - \psi_y\zeta_x}{\psi_x} + \zeta_y + \beta = K\sqrt{|\psi_x|}(\zeta_{xx} + \zeta_{yy}).$$

Example 3. The anticipated (potential) vorticity method was originally proposed by Sadourny and Basdevant [48]. The idea of this method is to approximate the diffusion effect in the vorticity equation by a weighted upwind estimate of the vorticity itself, i.e. by employing

$$\zeta^{\mathrm{a}}_t + J(\psi, \zeta^{\mathrm{a}}) = \nu J(\psi, \Delta^n J(\psi, \zeta^{\mathrm{a}})),$$

where ν is a constant, $n \in \mathbb{N}_0$ and ζ^{a} is the absolute vorticity. Here and in what follows $\Delta = \nabla^2$ is the two-dimensional Laplacian. The purpose of adding the specific forcing term on the right-hand side of the vorticity equation is to suppress the high frequency noise in the vorticity field and at the same time to ensure that energy is conserved during the integration while enstrophy is dissipated. The latter properties can be easily verified upon multiplying Eq. (2.3) with the stream function ψ and any function of the absolute vorticity ζ^{a}, respectively, and integrating over the domain Ω, see also [58].

There is a problem with this parameterization scheme in that it is not Galilean invariant. Galilean invariance (as well as the proper scale invariance), however, can be easily included by the method of invariantization. For the sake of demonstration, we consider the case of $n = 0$ here, which is the original version of the anticipated vorticity closure. Upon using the moving frame (2.10), we obtain

$$\iota(J(\psi, J(\psi, \zeta^{\mathrm{a}}))) = \frac{1}{\sqrt{|\psi_x|}} J(\psi_y, \zeta^{\mathrm{a}}) + \sqrt{|\psi_x|}\zeta^{\mathrm{a}}_{yy}.$$

Attaching this to the invariant representation of the vorticity equation (2.11), the vorticity equation with fully invariant closure reads ($\varepsilon = \operatorname{sgn}\psi_x$)

$$\zeta^{\mathrm{a}}_t + J(\psi, \zeta^{\mathrm{a}}) = \nu\sqrt{|\psi_x|}(\varepsilon J(\psi_y, \zeta^{\mathrm{a}}) + \psi_x\zeta^{\mathrm{a}}_{yy}). \tag{2.12}$$

It is obvious that this parameterization is quite different from that proposed in [48]. It cannot be brought in the form of nested Jacobian operators and it does not conserve energy any more (for the derivation of conservative invariant closure schemes, see Section 2.8). On the other hand, the inherent invariance of the closed vorticity equation (2.12) with respect to Galilean and scale symmetries is an appealing property for itself and might be relevant e.g. when vorticity dynamics is studied in a moving coordinate frame.

Quite recently, an approximate scale invariant formulation of the anticipated potential vorticity method was proposed in [15] using scale analysis techniques and physical reasoning. The motivation for this study is that modern weather and climate models might be required to operate on grids with variable resolution. Unfortunately, varying resolution in an atmospheric numerical model is not a simple task as most of the parameterization schemes employed are definitely not scale invariant, but rather tuned to yield best results on a fixed grid. Painful efforts might be necessary in order to adjust parameterization schemes of a numerical model to various spatial-temporal resolutions. Having a general method for deriving of scale-insensitive closure schemes at hand is therefore of potential practical interest in numerical geophysical fluid dynamics. Albeit simple, the method of invariantization of existing parameterization schemes may give appropriate closure schemes that are both physically meaningful and respect essential symmetries of a specific process to be represented numerically.

These are only two examples for fully invariant closure schemes. See one more example in the next section. In principle, each term of the form $S(I^1, \ldots, I^N)$, where S is a smooth function of its arguments and I^1, $\ldots$, I^N are differential invariants of G_1, satisfies the same requirement when added to the right hand side of Eq. (2.11). In other words, the general form of closure ansatzes for Eq. (2.11), which are invariant with respect to the entire group G_1, is

$$\zeta_t + \psi_x\zeta_y - \psi_y\zeta_x + \beta\psi_x = \psi_x S(I^1, \ldots, I^N).$$

2.7 Application of invariant parameterizations to turbulence modeling

In this section, we give an application in which we aim to demonstrate in practice the ideas outlined above and in [26]. This example deals with turbulence properties of the two-dimensional incompressible Euler equations. Strictly speaking, turbulence is a three-dimensional problem as a two-dimensional turbulent flow is not stable with respect to fully three-dimensional perturbations to that flow [49]. Nevertheless, there are countless studies concerning the turbulent properties of two-dimensional flow simply because it is a relevant problem in large-scale geophysical fluid dynamics, which behaves as approximately two-dimensional.

In short, the first theoretical results concerning two-dimensional turbulence were derived in [4, 29], following the pioneering work on three-dimensional turbulence done by Kolmogorov [28]. Extensive numerical studies have been carried out since then attempting to verify distinct aspects of the theory proposed [5, 6, 13, 22, 31]. The two-dimensional case is especially peculiar, as it admits infinitely many conservation laws including the conservation of energy. The energy in the barotropic vorticity is purely kinetic and can be represented in different ways using doubly periodic boundary conditions as

$$\mathcal{E} = \frac{1}{2}\int_\Omega \mathbf{v}^2 \mathrm{d}A = \frac{1}{2}\int_\Omega (\nabla\psi)^2 \mathrm{d}A = -\frac{1}{2}\int_\Omega \psi\zeta \mathrm{d}A, \tag{2.13}$$

where $\Omega = [0, L_x[\times [0, L_y[$ and $\mathrm{d}A = \mathrm{d}x\,\mathrm{d}y$. The special form of Eq. (2.3) leads to the following class of conservation laws

$$\mathcal{C}_g = \int_\Omega g(\zeta^\mathrm{a})\mathrm{d}A,$$

for any smooth function g of the absolute vorticity $\zeta^\mathrm{a} = \zeta + \mathsf{f}_0 + \beta y$. The most relevant realization of the above conservation laws in the present context is the enstrophy, given for the particular value $g = (\zeta^\mathrm{a})^2/2$.

First of all, consider the case of no differential rotation ($\beta = 0$), i.e. the Coriolis parameter f is approximated by the constant f_0, which is referred to as the f-plane approximation. It is the simultaneous conservation of energy and enstrophy in this case that leads to the remarkable behavior of two-dimensional turbulence [49, 57]. Starting with a random initial velocity (or stream function field), energy is transported to the large scale, while enstrophy is transported to the smaller scales. This cascade is associated with an organization of the vortices, with vortices of the same sign merging into bigger ones (though the precise mechanisms of the cascade including the role of the vortices are not yet fully understood). In order to initiate these fluxes of energy to the larger scale and enstrophy to the smaller scale and thus the process of organization, it is necessary to place a sink of enstrophy at the very small scales. This sink acts as a remover of enstrophy while ideally conserving energy, as the latter is transported away from the small scales on which the dissipation acts (which in practice is hard to realize in a numerical simulation using a finite number of grid points). It is believed that the form of the energy spectrum in a range above which dissipation is acting (inertial range) can be derived using scaling theory in a similar manner as it was shown by Kolmogorov for the three-dimensional case [49, 57].

The energy and enstrophy spectra $E(k)$ and $C(k)$ are defined by

$$\bar{\mathcal{E}} = \frac{1}{2L_xL_y}\int_\Omega \mathbf{v}^2\mathrm{d}A = \frac{1}{2L_xL_y}\int_\Omega (\nabla\psi)^2\mathrm{d}A = \int E(k)\mathrm{d}k,$$

$$\bar{\mathcal{C}} = \frac{1}{2L_xL_y}\int_\Omega \zeta^2\mathrm{d}A = \frac{1}{2L_xL_y}\int_\Omega (\Delta\psi)^2\mathrm{d}A = \int C(k)\mathrm{d}k,$$

where $\bar{\mathcal{E}}$ and $\bar{\mathcal{C}}$ are the average energy and average enstrophy, $k = \sqrt{(k^x)^2 + (k^y)^2}$ is the scalar wave number, k^x and k^y are the wave numbers in x- and y-direction, respectively. The possibility of using a single wave number is due to the assumption of isotropy that is generally made in turbulence theory and which is reasonable in the case of vanishing differential rotation [57]. According to the theory, the form of the energy spectrum in the inertial range should follow

$$E(k) \propto k^{-3}.$$

This is referred to as the *enstrophy cascade* in two-dimensional turbulence. Analogously, the *enstrophy spectrum* in the inertial range should follow

$$C_\mathrm{ens}(k) \propto k^{-1} = k^2E(k).$$

The impact of the beta-term in the vorticity equation on the turbulent cascades was first studied in [46]. In this seminal paper, it was remarked that the Rossby wave solutions admitted by the beta-plane equation can act as a source of anisotropization of turbulence at the larger scale. Qualitatively, at some stage the size of the vortices is big enough that they are exposed to the effect of differential rotation, which essentially hinders the tendency of vortex growth due

to the inverse energy cascade. Rather, the vortices evolve into Rossby wave and eventually to the formation of zonal jets as observed e.g. on giant planets. Depending on the precise setting used (e.g. strength of the differential rotation, additional energy injection to the system), the results of turbulence simulations can vary, but often energy spectra steeper than those predicted theoretically can be found [24, 32, 46].

In practice, the sink of enstrophy at the small scales is usually implemented by adding a hyperviscosity of the form

$$D = (-1)^{n-1}\nu\Delta^n\zeta \tag{2.14}$$

for $n \in \mathbb{N}^+$ to the right-hand side of Eq. (2.3), cf. Eq. (2.6). However, it can easily be checked that this form of hyperviscosity is not invariant under the Lie symmetry pseudogroups of the beta-plane and f-plane equations. More specifically, it violates the scale invariance of Eq. (2.3). *From the theoretical point of view, this violation appears to be especially odd, as it is precisely the scale invariance of the Euler equations that is used to derive the form of the energy spectrum in the inertial range.*

Theorem 2.1 directly implies that the invariantization $\iota(D) = (-1)^{n-1}\nu\sqrt{|\psi_x|^{2n-1}}\Delta^n\zeta$ is a differential invariant of the maximal Lie invariance pseudogroup of the vorticity equation. In view of the results of Section 2.6, we conclude that the form of the diffusion term obtained in the course of the invariantization is

$$\tilde{D} = |\psi_x|\iota(D) = (-1)^{n-1}\nu\sqrt{|\psi_x|^{2n+1}}\Delta^n\zeta.$$

The completely invariant formulation of the vorticity equation on the beta-plane with hyperdiffusion therefore reads

$$\zeta_t + \psi_x\zeta_y - \psi_y\zeta_x + \beta\psi_x = (-1)^{n-1}\nu\sqrt{|\psi_x|^{2n+1}}\Delta^n\zeta. \tag{2.15}$$

Note, however, that the price for introducing an invariant enstrophy sink is the *nonlinearity* of the (hyper)diffusion term. More generally, the situation is alike to the problem of finding a relation between the Reynolds stresses and the mean strain rate in the Reynolds averaged Navier–Stokes equations or in large–eddy simulations thereof. It was pointed out that establishing a relationship between the *nonlinear* Reynolds stresses and the *linear* strain rate (i.e. invoking the Boussinesq hypothesis) may lead to unrealistic results for certain turbulent flows such as in rotating or stratified fluids or those exposed to abrupt changes of the mean strain rate, see the discussions in [41, 59]. It is therefore worthwhile pointing out that the requirement of preserving the entire maximal Lie invariance pseudogroup of the barotropic vorticity equation on the beta-plane automatically yields nonlinear hyperdiffusion terms. For $n = 1$, the right-hand side of Eq. (2.15) can be considered as a generalized down-gradient parameterization for the eddy-vorticity flux, which is also a nonlinear quantity. That is, requiring a (hyper)diffusion scheme to be scale invariant, it is indispensable to use *nonlinear* (hyper)diffusion.

It is important to note that the anisotropic coefficient $\sqrt{|\psi_x|^{2n+1}}$ arises due to the special form of normalization conditions (2.9) we have chosen in Section 2.5 for the construction of the moving frame. This form is by no means unique but rather a consequence of the moving frame we have invoked. The situation is comparable to the discretization of differential equations, which can also be done in multiple ways. Some schemes have better properties than others and ultimately it is necessary to both analyze and test the various schemes for

different sets of problems. Having more than one possibility to construct invariant subgrid-scale schemes out of a given non-invariant scheme should therefore be considered as an advantage rather than as a drawback of the proposed method. The knowledge of the complete set of differential invariants, which is obtained as a byproduct when determining the invariantization map for a given group action, allows one to derive series of invariant closure schemes starting from that obtained as a direct result of the invariantization of the given initial scheme. This is facilitated by recombining a given invariant scheme using the differential invariants, as any functional combination of differential invariants is again a differential invariant.

A number of alternative (isotropic) forms of a completely invariant nonlinear hyperviscosity term for the vorticity equation on the beta-plane can therefore be suggested, e.g.

$$\tilde{D} = (-1)^{n-1}\nu\zeta^{2n+1}\Delta^n\zeta, \quad \tilde{D} = (-1)^{n-1}\nu\nabla(\zeta^{2n+1}\nabla\Delta^{n-1}\zeta), \quad \text{etc.},$$

which are derived upon recombining the differential invariants derived in Theorem 2.1. Due to the wide possibility for varying ansatzes for invariant parameterizations we can control different desirable conditions which proper invariant closure schemes should additionally satisfy, cf. Section 2.8.

Subsequently we will exclusively work with Eq. (2.15). Our motivation for choosing the anisotropic hyperdiffusion (2.15) rather than any of the above isotropic ones stems from recent experiments on turbulence which suggest that contrary to the Kolmogorov hypothesis the small scales might indeed feel the effects from the large scale being anisotropic, i.e. that anisotropy can propagate through to the very small scales, see e.g. [53]. However, future tests will be conducted so as to compare the different forms of invariant hyperdiffusion.

We give some numerical experiments using Eq. (2.15) and compare it with the respective non-invariant model that employs classical hyperdiffusion (2.14). Both models are integrated using a finite difference scheme and biharmonic dissipation is used in all the experiments, i.e. $n = 2$. The nonlinear terms on the left-hand side are discretized using the Arakawa Jacobian operator [3], which guarantees energy and enstrophy conservation of the spatial discretization in the case of vanishing dissipation, $\nu = 0$. A leapfrog scheme is used for the time stepping in conjunction with a Robert–Asselin–Williams filter [60], in order to suppress the computational mode. The size of the domain is $L_x = L_y = 2\pi$, with a default of $N = 1024$ grid points in each direction, $\beta = 1$. The initial condition is a Gaussian random stream function field, with the initial energy spectrum given by the function $E(k) \propto k^3 \exp(-3k^2/k_p^2)$, where $k_p = 64$. No normalization of the initial energy was used. The value of ν was chosen to be $\nu_{\text{inv}} = 1 \cdot 10^{-10}$ in the invariant case and $\nu_{\text{ninv}} = 2 \cdot 10^{-9}$ for the non-invariant simulations. Note that the value of ν_{ninv} has been selected to lie in between the values given in [13] for the two integrations using 512^2 and 4096^2 grid points. The value of ν_{inv} has been chosen so that $\nu_{\text{inv}} \approx \max(\nu_{\text{ninv}}\sqrt{|\psi_x|^5})$ initially for the sake of comparison.

Both models have been integrated for approximately 10 000 time steps using $\Delta t = 1 \cdot 10^{-3}$. Hence, all the results presented below were evaluated at approximately $t = 10$, which should be long enough so that inertial ranges can form in the energy and enstrophy spectra. Below, we shall like to present the enstrophy spectra for fully developed freely decaying turbulence using both the invariant and the non-invariant hyperdiffusion terms. As was said above, according to the Batchelor–Kraichnan theory the enstrophy spectrum should be of the form k^{-1} in the intertial range. However, finding experimental evidence for a spectrum of

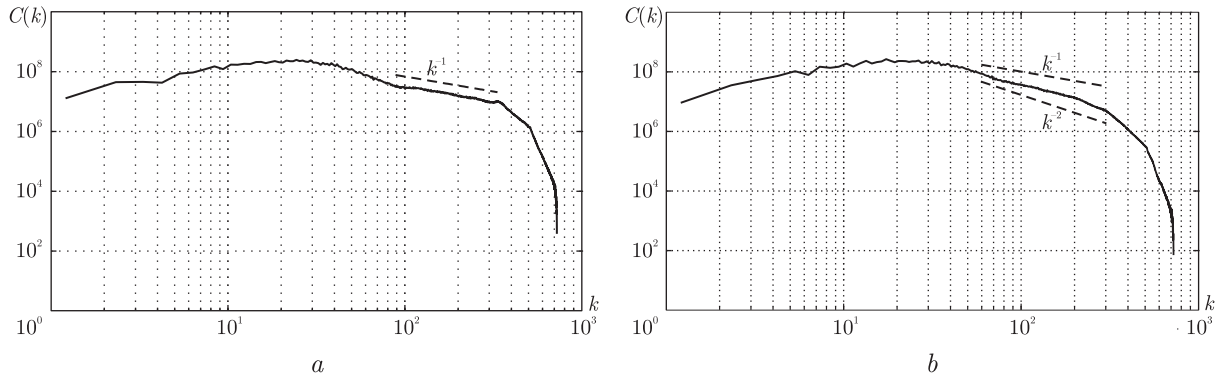

Figure 2.1: Enstrophy spectrum at approximately $t = 10$ using (a) invariant hyperdiffusion and (b) non-invariant hyperdiffusion.

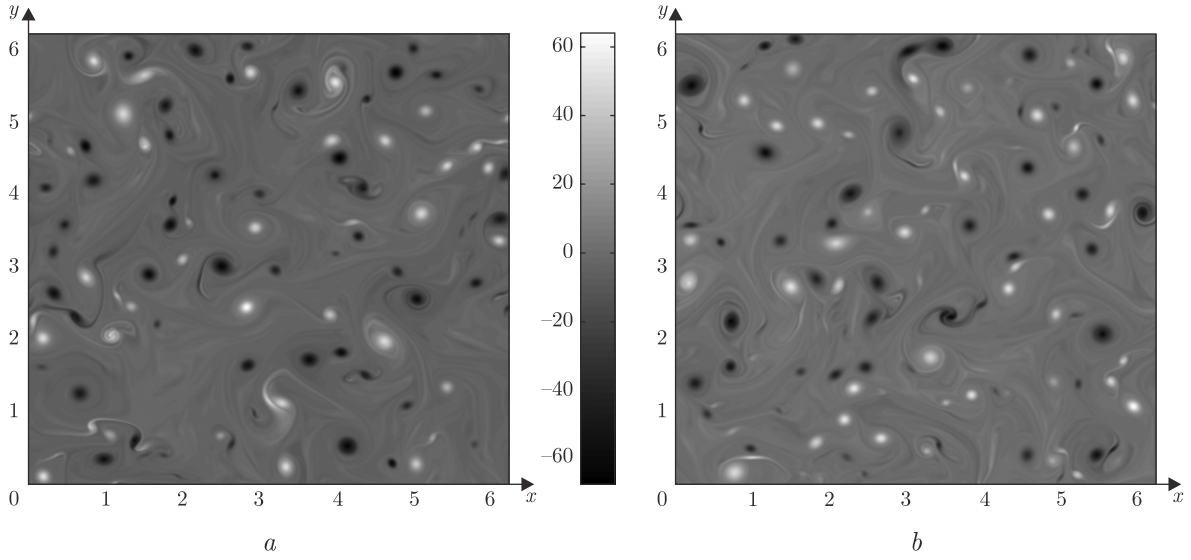

Figure 2.2: Vorticity field at approximately $t = 10$ using (a) invariant hyperdiffusion and (b) non-invariant hyperdiffusion.

this form proved rather hard and most numerical simulations carried out so far yield steeper spectra.

In Fig. 2.1a we show the enstrophy spectrum found from the simulation using invariant hyperdiffusion. In the region between approximately $k = 100$ up to $k = 300$ the spectrum follows k^{-1} almost perfectly. That is, the invariant hyperdiffusion of the form used in (2.15) leads to an experimental verification of the Batchelor–Kraichnan theory.

In Fig. 2.1b we show the corresponding enstrophy spectrum obtained using conventional (non-invariant) hyperdiffusion. As in the majority of turbulence simulations, also we obtain a spectrum in the inertial range that is *steeper* than k^{-1}, lying between k^{-1} and k^{-2}, in this case. Moreover, it is instructive to note that the lower parts of the spectra (up to the respective inertial ranges) are rather similar for both schemes, while differences occur within the inertial and in the diffusion ranges. This observation underpins that the proposed nonlinear invariant hyperdiffusion is physically acting as a viscousity term in Eq. (2.15).

Fig. 2.2 shows the associated vorticity fields obtained using the invariant and non-invariant hyperdiffusion schemes at the end of the integration. Note that the value of β chosen is rather

small (and much smaller as compared to the value of $\beta = 3$ used in [32]) so the effects of differential rotation on the vorticity fields are rather minimal. Both fields look qualitatively similar verifying that invariant hyperdiffusion is capable of producing a physically meaningful vorticity field.

Remark 2.2. Decaying turbulence simulations are an important class of tests for numerical integration schemes. On the other hand, from the point of view of both the theory and application, it is generally more instructive when Eq. (2.3) is augmented with some forcing which supplies energy to the system and thereby prevents turbulence from dying out. As it is then usually necessary to damp out the energy which is otherwise piling up at small wave numbers (large scales) due to the inverse energy cascade, an additional drag term is introduced in Eq. (2.3). This drag term can be either physical (e.g. linear Ekman drag due to bottom friction) or, similar as hyperviscosity, scale selective. In the latter case, one uses a *hypoviscosity* [16], which is given by adding a term proportional to $\Delta^{-n}\zeta$, which acts scale selective by emphasizing the large scale and thus is effectively energy removing. Again, one could raise the question whether such a hypofriction should possess some invariance properties, but this is beyond the scope of the present paper and should be considered in a forthcoming study.

2.8 Conservative invariant parameterizations

A parameterization is called *conservative* if the corresponding closed system of differential equations possesses nonzero conservation laws. Special attention should be paid to parameterizations possessing conservation laws that have a clear physical interpretation (such as the conservation of energy, mass, momentum, etc.) and that originate from the conservation laws of the initial system of equations. If a parameterization is both conservative and invariant with respect to a pseudogroup of transformations, it is called a *conservative invariant parameterization*.

The general method for singling out conservative parameterizations among invariant closure ansatzes is based on the usage of the Euler operators, i.e. variational derivatives with respect to the dependent variables [42]. Suppose that $\tilde{\mathcal{L}}_\theta$: $\tilde{L}^l(x, \bar{u}_{(n)}, \theta) = 0$, $l = 1, \dots, m$, $\theta = \theta(I^1, \dots, I^N)$ represent a family of local parameterizations for a system $\mathcal{L}$: $L^l(x, u_{(n)}) = 0$, $l = 1, \dots, m$, which are invariant with respect to a pseudogroup G. Here $\tilde{L}^l$ are fixed smooth functions of their arguments. The tuple θ of arbitrary elements consists of smooth functions of certain differential invariants I^1, $\dots$, I^N of G. It runs through a set of such tuples constrained by a system of differential equations, where I^1, $\dots$, I^N play the role of independent variables. We require the tuples $(\lambda^{m1}, \dots, \lambda^{ml})$, $m = 1, \dots, M$, of differential functions of u to be characteristics of M linearly independent local conservation laws of the system $\tilde{\mathcal{L}}_\theta$ for some values of θ, i.e. for each m the combination $\lambda^{m1}\tilde{L}^1 + \dots + \lambda^{ml}\tilde{L}^l$ is a total divergence. The theorem on characterization of total divergences [42, Theorem 4.7] then implies the equations $\mathsf{E}^a(\lambda^{m1}\tilde{L}^1 + \dots + \lambda^{ml}\tilde{L}^l) = 0$ for each $m = 1, \dots, M$ and $a = 1, \dots, q$, where E^a is the Euler operator associated with the dependent variable u^a, $\mathsf{E}^a f = \sum_\alpha (-\mathrm{D})^\alpha f_{u^a_\alpha}$. Splitting these equations with respect to derivatives of u wherever this is possible, one constructs the system of determining equations with respect to θ, which should be solved in order to derive the corresponding conservative invariant parameterizations.

As the direct computation is too cumbersome, we use some heuristic arguments and look for a diffusion ansatz for the barotropic vorticity equation on the beta-plane that satisfies the following relevant and valuable conditions:

- It is invariant with respect to the entire maximal Lie invariance pseudogroup G_1 of Eq. (2.3).
- The subgrid-scale term or, more generally, the sink term to be represented is a differential function of the vorticity ζ (namely, a polynomial depending only on derivatives of ζ with respect to the space variables x and y).
- This expression is as similar as possible to the hyperviscosity term (2.14).
- And, last but not least, the parameterization is conservative. More precisely, it possesses all the conservation laws of Eq. (2.3) with zero-order characteristics.

The second point guarantees the invariance of the corresponding diffusion ansatz under all transformations from G_1 that do not involve scalings. In order to provide the scale invariance, we should just balance the scaling weights of derivatives of ζ in the diffusion term. Moreover, these derivatives should be composed in such a way that allows integrating by parts in order to represent the diffusion term multiplied by an arbitrary zero-order conservation-law characteristic of Eq. (2.3) in conserved form. An example of such a parameterization is given by

$$\zeta_t+\psi_x\zeta_y-\psi_y\zeta_x+\beta\psi_x=D,\quad D=\nu\Delta\frac{\Delta\zeta^7}{\zeta}=7\nu\Delta(\zeta^5\Delta\zeta+6\zeta^4(\nabla\zeta)^2). \tag{2.16}$$

All the properties listed above can be checked for the sink term (2.16). Thus, the expression for D from (2.16) involves only the vorticity and its derivatives and is quite similar to (2.14). Moreover, the diffusion D is a globally defined differential function which is a polynomial of its arguments. The invariance of Eq. (2.16) with respect to G_1 can be simply checked using the infinitesimal invariance criterion. A more sophisticated way to check this invariance is to rewrite Eq. (2.16) in terms of normalized invariants of the pseudogroup G_1, which will not be done explicitly here. As an unexpected but valuable bonus we have that the maximal Lie symmetry pseudogroup of Eq. (2.16) with the same term D in the case of the f-plane ($\beta=0$) is even wider than G_1. It also includes the usual rotations of the variables (x,y) and the generalized Galilean boosts in y-direction, which belong to the Lie symmetry pseudogroup G_0 of the barotropic vorticity equation on the f-plane. This in particular means that the parameterization (2.16) is isotropic.

The space of zero-order characteristics of Eq. (2.3) is generated by the characteristics $\lambda=f(t)$, $\lambda=g(t)y$ and $\lambda=\psi$, where f and g run through the set of smooth functions of t. The physically most important of these characteristics are $\lambda=1$, $\lambda=y$ and $\lambda=\psi$, which are associated with the conservation of circulation, x-momentum and energy. Any zero-order characteristic of Eq. (2.3) is a characteristic of Eq. (2.16). Indeed, denoting

$$L:=\zeta_t+\psi_x\zeta_y-\psi_y\zeta_x+\beta\psi_x-D$$

we derive that

$$\begin{aligned}
fL&=\mathrm{D}_x\left(f\psi_{xt}+f\psi\zeta_y+f\beta\psi-\nu f\mathrm{D}_x\frac{\Delta\zeta^7}{\zeta}\right)+\mathrm{D}_y\left(f\psi_{yt}-f\psi\zeta_x-\nu f\mathrm{D}_y\frac{\Delta\zeta^7}{\zeta}\right),\\
gyL&=\mathrm{D}_x\left(gy\psi_{xt}+gy\psi\zeta_y-\frac{g}{2}(\psi_y)^2+gy\beta\psi-\nu gy\mathrm{D}_x\frac{\Delta\zeta^7}{\zeta}\right)\\
&\quad+\mathrm{D}_y\left(gy\psi_{yt}-g\psi_y-gy\psi\zeta_x+g\psi\psi_{xy}-\nu gy\mathrm{D}_y\frac{\Delta\zeta^7}{\zeta}+\nu g\frac{\Delta\zeta^7}{\zeta}\right),
\end{aligned}$$

$$\psi L = \mathrm{D}_t\left(-\frac{1}{2}(\nabla\psi)^2\right) + \mathrm{D}_x\left(\psi\psi_{xt} + \frac{1}{2}\psi^2\zeta_y + \frac{\beta}{2}\psi^2 - \nu\psi\mathrm{D}_x\frac{\Delta\zeta^7}{\zeta} + \nu\psi_x\frac{\Delta\zeta^7}{\zeta} - \nu\mathrm{D}_x\zeta^7\right)$$
$$+ \mathrm{D}_y\left(\psi\psi_{yt} - \frac{1}{2}\psi^2\zeta_x - \nu\psi\mathrm{D}_y\frac{\Delta\zeta^7}{\zeta} + \nu\psi_y\frac{\Delta\zeta^7}{\zeta} - \nu\mathrm{D}_y\zeta^7\right).$$

If we grant that the vorticity equation coupled with some diffusive term possesses a smaller number of conservation laws (e.g. owing to the special physical properties of this diffusion), we can use a simpler form for the expression D. For example, the differential function $D = \nu\Delta\zeta^4$ leads to a parameterization which is invariant with respect to the entire pseudogroup G_1 and possesses conservation laws with the characteristics $\lambda = f(t)$, $\lambda = g(t)y$ for arbitrary values of the smooth parameter-functions f and g.

The parameterization (2.16) demonstrates the feasibility of combining invariant and conservative properties of closure schemes. This possibility is important for two obvious reasons. Firstly, conservation laws incorporate relevant physical information that is worth being preserved by a parameterization scheme. Secondly, from the point of view of constructing parameterization schemes, the requirement of preserving both symmetries and conservation laws leads to a more specific class of schemes than considering either only symmetries or only conservation laws. The additional narrowing of the class of admitted schemes using geometric constraints can then help to reduce the number of schemes that must be tested numerically so as to find the optimal parameterization for a given process.

2.9 Conclusion and discussion

The differential invariants of the Lie symmetry pseudogroup G_1 of the barotropic vorticity equation on the beta-plane are computed using the technique of moving frames for Lie pseudogroups. A basis of these differential invariants along with the associated operators of invariant differentiation is established. Together, they serve to completely describe the algebra of differential invariants of G_1. Although differential invariants have many applications (such as the integration of ordinary differential equations [42], computation of so-called differentially invariant solutions [23, 30], the construction of invariant numerical discretization schemes [6], etc.), in the paper we focus on their usage in the construction of invariant closure schemes or, perhaps more generally, invariant diffusion terms for the averaged vorticity equation. This is one of the two general methods proposed in [26] to derive parameterization schemes with symmetry properties. As an alternative to the direct usage of elementary differential invariants that can be build together to yield invariant closure schemes, we propose the method of invariantization of existing parameterization schemes. This method is along the line of invariantization of existing discretization schemes as introduced in [12, 22]. Although this method is straightforward to apply, a potential complication is that the result depends on the particular choice of the moving frame and therefore does not lead to a unique invariant counterpart of an existing non-invariant scheme. As a consequence, it might be necessary to modify invariantized closure schemes and to test different invariantizations in order to devise physically valuable closures.

The differential invariants derived are used to construct invariant hyperdiffusion terms in order to model the behavior of two-dimensional freely decaying turbulence. The resulting enstrophy spectrum exhibits an arc of approximate k^{-1} slope which is the theoretically derived shape for the postulated enstrophy inertial range. It should be stressed, though, that the obtained enstrophy spectrum should be taken with a pinch of salt. Since the derivation of the

theoretical form of the spectra in [4, 29] it has been tried in numerous studies to obtain these spectra in numerical simulations. Although results often vary, spectra are found with a steeper slope than the predicted k^{-1} curve as described in [5, 6, 13, 30, 32, 52]. It seems to be generally agreed today that the presence of the stable coherent vortices, which is the main feature of two-dimensional turbulence, has a strong impact on the derived enstrophy spectra. This holds in the case of turbulence both on the f-plane and on the beta-plane. The introduction of an invariant hyperdiffusion-like term certainly complicates the situation as diffusion then is coupled *nonlinearly* to the vorticity equation. On the other hand, it was indicated that the presence of the beta-term in the vorticity equation allows for a nonlocal transfer of anisotropy from the larger to the smaller scales [32]. A nonlinear diffusion term has the potential to support such a nonlocal scale interaction and thereby serves as a potential parameterization scheme for numerical models. It should be stressed in this context that in all the simulations we have carried out, the rate of energy dissipation was lower than using classical hyperdiffusion even in quite low-resolution numerical experiments.

Apart from the discussion above, the possibility of constructing hyperdiffusion-like enstrophy sink terms that lead to scale invariant enstrophy spectra seems to be a valuable property for itself. It is precisely the scale invariance of the Euler equations that is used to predict the behavior of two-dimensional turbulence in the inertial range and therefore the availability of dissipative versions of the vorticity equation having the same invariance properties as the inviscid vorticity equation might be a general advantage. Heuristically, one can expect that an invariant closure scheme should be better adapted for the problem of reproducing features that have been derived using symmetries (as the isotropic enstrophy spectrum), similarly as an invariant discretization scheme often reproduces better invariant exact solutions of a differential equation than non-invariant discretization schemes [21]. This assumption is supported by the proved relevance of Lie symmetries in turbulence theory [35]. The results obtained in the present paper do not contradict this assumption, keeping in mind especially that the premises invoked to obtain the theoretical form of the spectra are at present under revision. In this context, it should again be stressed that there is a multitude of invariant parameterization schemes or invariant diffusion terms that can be coupled to the vorticity equation on the beta-plane. The fact that already the simplest invariantized version (2.15) of the hyperdiffusion term (which has obvious weaknesses) shows quite good properties in the course of our numerical tests is a motivating result which is worth pointing out. Nevertheless, in order to verify and better assess the ability of invariant hyperdiffusion schemes to model turbulence on the beta-plane, further theoretical and numerical studies must be carried out.

The method we propose in this paper is fully generalizable. It is the number of variables of a model and its symmetry group that determine whether the method is computationally more complicated to realize, but this complication is not conceptual. Thus, the relative simplicity of constructing diffusion schemes that are invariant under the entire maximal Lie invariance group is a particular feature of the beta-plane vorticity equation, which is computationally more involved for vorticity dynamics on the f-plane. The complication with the latter model is that the corresponding maximal Lie invariance pseudogroup G_0 is even wider than G_1. This makes it much harder to derive reasonably simple closure schemes that are invariant under the entire pseudogroup G_0, see the discussion in [26], where a generating set of differential invariants of G_0 and a complete set of its independent operators of invariant differentiation are determined. A possible remedy for this complication is to consider closure schemes that are invariant only

under certain subgroups of the maximal Lie invariance pseudogroup of the f-plane equation. As highlighted in the present paper, the selection of such subgroups can be justified for physical reasons when boundaries come into play.

Another novel feature of the present paper is the explicit inclusion of conservation laws in invariant closure schemes. The chance of constructing such conservative invariant parameterization schemes is of obvious physical relevance. For physical processes that do not violate particular conservation laws, it is natural to require the associated parameterization to be also conservative. It was demonstrated in the paper for the vorticity equation on the beta-plane that the concepts of invariant and conservative parameterization schemes can be united to yield closure ansatzes that preserve both all the symmetries and certain conservation laws of this equation. The construction of further invariant conservative closure schemes as well as their exhaustive testing will be a next major challenge in the application of ideas of group analysis to the parameterization problem.

Acknowledgements

Some of the numerical simulations were carried out using the infrastructure of the Institute of Meteorology and Geophysics of the University of Vienna. AB acknowledges the assistance of Professor Leopold Haimberger in carrying out these simulations. The authors are grateful to Jean-Christophe Nave, Artur Sergyeyev, Matthias Sommer, Miguel Teixeira, Pavel Winternitz and Jun-Ichi Yano for valuable discussions. We thank the two anonymous reviewers for their helpful suggestions and comments, which led to an improved version of this manuscript. This research was supported by the Austrian Science Fund (FWF), projects J3182–N13 (AB), P20632 (ROP) and P25064 (EDSCB). The present work is carried out within the framework of COST Action ES0905.

2.A Symmetries of the vorticity equation on the beta-plane

We aim to detail the computation of the maximal Lie invariance algebra $\mathfrak{g}_1$ of the vorticity equation (2.3) here. Full expositions on finding Lie symmetries of differential equations can be found in the standard textbooks [1, 4, 30, 42]. More details on the symmetries (and exact solutions) of the vorticity equation are presented in [5].

Given a generator

$$Q = \tau(t,x,y,\psi)\partial_t + \xi(t,x,y,\psi)\partial_x + \eta(t,x,y,\psi)\partial_y + \varphi(t,x,y,\psi)\partial_\psi. \tag{2.17}$$

of a one-parameter point symmetry group of the vorticity equation

$$\Delta = \zeta_t + \psi_x\zeta_y - \psi_y\zeta_x + \beta\psi_x = 0, \quad \zeta = \psi_{xx} + \psi_{yy},$$

the infinitesimal invariance criterion [30, 42] implies $Q_3(\Delta) = 0$, which has to hold on the manifold $\Delta = 0$, where Q_3 denotes the third prolongation of the vector field Q. Explicitly, the prolonged vector field Q_3 is defined by $Q_3 = Q + \sum_{0<|\alpha|\leqslant 3}\varphi^\alpha\partial_{\psi_\alpha}$ and the coefficients of Q_3 are derived from the general prolongation formula,

$$\varphi^\alpha = \mathrm{D}_t^{\alpha_1}\mathrm{D}_x^{\alpha_2}\mathrm{D}_y^{\alpha_3}(\varphi - \tau\psi_{\delta_1} - \xi\psi_{\delta_2} - \eta\psi_{\delta_3}) + \tau\psi_{\alpha+\delta_1} + \xi\psi_{\alpha+\delta_2} + \eta\psi_{\alpha+\delta_3}. \tag{2.18}$$

Here we use the notation introduced in the beginning of Section 2.5. Then the condition $Q_3(\Delta) = 0$ expands to

$$\varphi^{120} + \varphi^{102} + \varphi^{010}\zeta_y + \psi_x(\varphi^{021} + \varphi^{003}) - \varphi^{001}\zeta_x - \psi_y(\varphi^{030} + \varphi^{012}) + \beta\varphi^{010} = 0,$$

and the constraint that $Q_3(\Delta) = 0$ has to hold only on the manifold of $\Delta = 0$ is taken into account by substituting $\psi_{txx} = -\psi_{tyy} - \psi_x\zeta_y + \psi_y\zeta_x - \beta\psi_x$ wherever ψ_{txx} occurs. As the coefficients of Q are only functions of t, x, y and ψ, the expanded condition can be split with respect to the various derivatives of ψ. This splitting yields the determining equations for the coefficients of the vector field Q,

$$\begin{aligned} &\tau_x = \tau_y = \tau_\psi = \xi_y = \xi_\psi = \eta_t = \eta_x = \eta_\psi = \varphi_x = 0,\\ &\xi_x = \eta_y = -\tau_t, \quad \varphi_y = -\xi_t, \quad \varphi_\psi = -3\tau_t. \end{aligned} \tag{2.19}$$

The general solution of this system of determining equations reads

$$\tau = c_1 t + c_2, \quad \xi = -c_1 x + \tilde{f}(t), \quad \eta = -c_1 y + c_3, \quad \varphi = -3c_1\psi - \tilde{f}_t y + \tilde{g}(t),$$

where $\tilde{f}$ and $\tilde{g}$ run through the set of smooth functions of t. Thus, the maximal Lie invariance algebra of infinitesimal symmetries of the barotropic vorticity equation on the beta-plane is spanned by the vector fields

$$\mathcal{D} = t\partial_t - x\partial_x - y\partial_y - 3\psi\partial_\psi, \quad \partial_t, \quad \partial_y, \quad \mathcal{X}(\tilde{f}) = \tilde{f}(t)\partial_x - \tilde{f}_t(t)y\partial_\psi \quad \mathcal{Z}(\tilde{g}) = \tilde{g}(t)\partial_\psi.$$

2.B Algebra of differential invariants for the vorticity equation

In this appendix we present the details for the proof of Theorem 2.2 which exhaustively describes the algebra of differential invariants for the maximal Lie invariance pseudogroup of the barotropic vorticity equation on the beta-plane.

A complete set of independent operators of invariant differentiation is derived by invariantization of the usual operators of total differentiation, yielding

$$\mathrm{D}^{\mathrm{i}}_t = \frac{1}{\sqrt{|\psi_x|}}(\mathrm{D}_t - \psi_y\mathrm{D}_x), \quad \mathrm{D}^{\mathrm{i}}_x = \sqrt{|\psi_x|}\mathrm{D}_x, \quad \mathrm{D}^{\mathrm{i}}_y = \sqrt{|\psi_x|}\mathrm{D}_y. \tag{2.20}$$

This is practically realized via substituting the expressions (2.10) for the pseudogroup parameters into the implicit differentiation operators (2.8). Any operator of invariant differentiation related to the pseudogroup G_1 is locally a combination of the operators (2.20) with functional coefficients depending only on differential invariants of G_1. The commutation relations between the operators $\mathrm{D}^{\mathrm{i}}_t$, $\mathrm{D}^{\mathrm{i}}_x$ and $\mathrm{D}^{\mathrm{i}}_y$ are

$$\begin{aligned} &[\mathrm{D}^{\mathrm{i}}_t, \mathrm{D}^{\mathrm{i}}_x] = \frac{\varepsilon}{2}I_{020}\mathrm{D}^{\mathrm{i}}_t + \Big(I_{011} + \frac{\varepsilon}{2}I_{110}\Big)\mathrm{D}^{\mathrm{i}}_x,\\ &[\mathrm{D}^{\mathrm{i}}_t, \mathrm{D}^{\mathrm{i}}_y] = \frac{\varepsilon}{2}I_{011}\mathrm{D}^{\mathrm{i}}_t + I_{002}\mathrm{D}^{\mathrm{i}}_x + \frac{\varepsilon}{2}I_{110}\mathrm{D}^{\mathrm{i}}_y,\\ &[\mathrm{D}^{\mathrm{i}}_x, \mathrm{D}^{\mathrm{i}}_y] = \frac{\varepsilon}{2}I_{020}\mathrm{D}^{\mathrm{i}}_y - \frac{\varepsilon}{2}I_{011}\mathrm{D}^{\mathrm{i}}_x. \end{aligned} \tag{2.21}$$

In order to completely describe the algebra of differential invariants of G_1, it remains to establish a basis of differential invariants such that any differential invariant of G_1 can be represented as a function of basis elements and their invariant derivatives. It is also necessary to

compute a complete system of syzygies between basis invariants. For this aim, we will evaluate the recurrence relations between the normalized differential invariants and the differentiated differential invariants as detailed in [9, 19]. The starting point for the application of the general algorithm to the maximal Lie invariance pseudogroup G_1 of the vorticity equation on the beta-plane is the system of determining equations for the coefficients of a vector field (2.17) from the maximal Lie invariance algebra of Eq. (2.3), which is given through system (2.19). Consider the prolonged operator $Q_\infty = Q + \sum_{|\alpha|>0} \varphi^\alpha \partial_{\psi_\alpha}$. The coefficients of Q_∞ are calculated by the standard prolongation formula (2.18). In view of the determining equations, the coefficients φ^α take the form

$$\varphi^\alpha = (\alpha_2+\alpha_3-\alpha_1-3)\tau_t\psi_\alpha - \sum_{k=1}^{\alpha_1}\binom{\alpha_1}{k}\xi_{(k)}\psi_{\alpha-k\delta_1+\delta_2} + \left\{\begin{array}{ll} -\xi_{(\alpha_1+1)}, & \alpha_2=0,\ \alpha_3=1 \\ \varphi_{(\alpha_1)}, & \alpha_2=\alpha_3=0 \end{array}\right\},$$

where $\xi_{(k)} = \partial^k\xi/\partial t^k$ and $\varphi_{(k)} = \partial^k\varphi/\partial t^k$, $k = 0, 1, 2, \dots$. We collect the coefficients of Q and their derivatives appearing in the expressions for the prolonged coefficients of Q and denote the associated invariantized objects, which are differential forms, as $\hat\tau^0 = \iota(\tau)$, $\hat\tau^1 = \iota(\tau_t)$, $\hat\xi^k = \iota(\xi_{(k)})$, $\hat\eta = \iota(\eta)$ and $\hat\varphi^k = \iota(\varphi_{(k)})$. In the course of the normalization (2.9) the invariantized counterparts $\hat\varphi^\alpha = \iota(\varphi^\alpha)$ of the prolonged coefficients of Q are

$$\hat\varphi^{j00} = \hat\varphi^j - \varepsilon\hat\xi^j - \sum_{k=1}^{j-1}\binom{j}{k}I_{j-k,10}\hat\xi^k \quad \text{if} \quad j>0, \quad \hat\varphi^{j01} = -\hat\xi^{j+1} - \sum_{k=1}^{j}\binom{j}{k}I_{j-k,11}\hat\xi^k,$$

$$\hat\varphi^\alpha = (\alpha_2+\alpha_3-\alpha_1-3)I_\alpha\hat\tau^1 - \sum_{k=1}^{\alpha_1}\binom{\alpha_1}{k}I_{\alpha-k\delta_1+\delta_2}\hat\xi^k \quad \text{if} \quad \alpha_2>0 \quad \text{or} \quad \alpha_3>1.$$

For lower values of $|\alpha|$, $0<|\alpha|\leqslant 3$, we calculate

$$\begin{aligned}
&\hat\varphi^{100} = \hat\varphi^1 - \varepsilon\hat\xi^1, \quad \hat\varphi^{010} = -2\hat\tau^1, \quad \hat\varphi^{001} = -\hat\xi^1,\\
&\hat\varphi^{200} = \hat\varphi^2 - \varepsilon\hat\xi^2 - 2I_{110}\hat\xi^1, \quad \hat\varphi^{110} = -3I_{110}\hat\tau^1 - I_{020}\hat\xi^1, \quad \hat\varphi^{101} = -\hat\xi^2 - I_{011}\hat\xi^1,\\
&\hat\varphi^{020} = -I_{020}\hat\tau^1, \quad \hat\varphi^{011} = -I_{011}\hat\tau^1, \quad \hat\varphi^{002} = -I_{002}\hat\tau^1,\\
&\hat\varphi^{300} = \hat\varphi^3 - \varepsilon\hat\xi^3 - 3I_{110}\hat\xi^2 - 3I_{210}\hat\xi^1,\\
&\hat\varphi^{210} = -4I_{210}\hat\tau^1 - I_{020}\hat\xi^2 - 2I_{120}\hat\xi^1, \quad \hat\varphi^{201} = -\hat\xi^3 - I_{011}\hat\xi^2 - 2I_{111}\hat\xi^1,\\
&\hat\varphi^{120} = -2I_{120}\hat\tau^1 - I_{030}\hat\xi^1, \quad \hat\varphi^{111} = -2I_{111}\hat\tau^1 - I_{021}\hat\xi^1, \quad \hat\varphi^{102} = -2I_{102}\hat\tau^1 - I_{012}\hat\xi^1,\\
&\hat\varphi^{030} = \hat\varphi^{021} = \hat\varphi^{012} = \hat\varphi^{003} = 0.
\end{aligned}$$

From the recurrence relations for the phantom invariants $H^0 = \iota(t)$, $H^1 = \iota(x)$, $H^2 = \iota(y)$, $I_{i00} = \iota(\psi_{i00})$, $I_{i01} = \iota(\psi_{i01})$, $i = 0, 1, \dots$, and $I_{010} = \iota(\psi_{010})$, which are

$$\mathrm{d_h}H^0 = \omega^1 + \hat\tau^0 = 0, \quad \mathrm{d_h}H^1 = \omega^2 + \hat\xi^0 = 0, \quad \mathrm{d_h}H^2 = \omega^3 + \hat\eta = 0, \quad \mathrm{d_h}I_{000} = \omega^2 + \hat\varphi^0 = 0,$$

$$\mathrm{d_h}I_{j00} = I_{j10}\omega^2 + \hat\varphi^j - \varepsilon\hat\xi^j - \sum_{k=1}^{j-1}\binom{j}{k}I_{j-k,10}\hat\xi^k = 0, \quad j = 1, 2, \dots,$$

$$\mathrm{d_h}I_{j01} = I_{j11}\omega^2 + I_{j02}\omega^3 - \hat\xi^{j+1} - \sum_{k=1}^{j}\binom{j}{k}I_{j-k,11}\hat\xi^k = 0, \quad j = 0, 1, \dots,$$

$$\mathrm{d_h}I_{010} = I_{110}\omega^1 + I_{020}\omega^2 + I_{011}\omega^3 - 2\hat\tau^1 = 0,$$

where $\omega^1 = \iota(\mathrm{d}t)$, $\omega^2 = \iota(\mathrm{d}x)$ and $\omega^3 = \iota(\mathrm{d}y)$, we derive expressions for the invariantized Maurer–Cartan forms

$$\hat\tau^0 = -\omega^1, \quad \hat\xi^0 = -\omega^2, \quad \hat\eta = -\omega^3, \quad \hat\varphi^0 = -\omega^2, \quad \hat\tau^1 = \tfrac12(I_{110}\omega^1 + I_{020}\omega^2 + I_{011}\omega^3),$$
$$\hat\xi^j = I_{j-1,11}\omega^2 + I_{j-1,02}\omega^3 - \sum_{k=1}^{j-1}\binom{j-1}{k} I_{j-k-1,11}\hat\xi^k,$$
$$\hat\varphi^j = -I_{j10}\omega^2 + \varepsilon\hat\xi^j + \sum_{k=1}^{j-1}\binom{j}{k} I_{j-k,10}\hat\xi^k,$$

$j = 1, 2, \ldots$. The forms $\hat\xi^j$ should be calculated recursively starting from $j = 1$. Thus,

$$\hat\xi^1 = I_{011}\omega^2 + I_{002}\omega^3,$$
$$\hat\xi^2 = (I_{111} - I_{011}^2)\omega^2 + (I_{102} - I_{011}I_{002})\omega^3,$$
$$\hat\xi^3 = (I_{211} - 3I_{011}I_{111} + I_{111}^3)\omega^2 + (I_{202} - 3I_{011}I_{102} + I_{011}^2 I_{002})\omega^3, \quad \ldots \quad .$$

In general, $\hat\xi^j = \hat\xi^{j,2}\omega^2 + \hat\xi^{j,3}\omega^3$, where the coefficients $\hat\xi^{j,2}$ and $\hat\xi^{j,3}$ are expressed in terms of normalized invariants I_α with $|\alpha| \leqslant j+1$.

The recurrence relations for non-phantom normalized invariants correspondingly read

$$\mathrm{d_h} I_{\alpha_1\alpha_2\alpha_3} = I_{\alpha+\delta_1}\omega^1 + I_{\alpha+\delta_2}\omega^2 + I_{\alpha+\delta_3}\omega^3 + (\alpha_2+\alpha_3-\alpha_1-3)I_\alpha\hat\tau^1 - \sum_{k=1}^{\alpha_1}\binom{\alpha_1}{k} I_{\alpha-k\delta_1+\delta_2}\hat\xi^k \quad \text{if} \quad \alpha_2 > 0 \quad \text{or} \quad \alpha_3 > 1.$$

As by definition $\mathrm{d_h}F = (\mathrm{D}_t^{\mathrm{i}}F)\omega^1 + (\mathrm{D}_x^{\mathrm{i}}F)\omega^2 + (\mathrm{D}_y^{\mathrm{i}}F)\omega^3$, the above recurrence relations can be split into a list of equations for first-order invariant derivatives of normalized differential invariants I_α with $\alpha_2 > 0$ or $\alpha_3 > 1$ by taking into account the expressions for the invariantized Maurer–Cartan forms:

$$\begin{aligned}
&\mathrm{D}_t^{\mathrm{i}} I_\alpha = I_{\alpha+\delta_1} + \frac{\alpha_2+\alpha_3-\alpha_1-3}{2} I_{110}I_\alpha,\\
&\mathrm{D}_x^{\mathrm{i}} I_\alpha = I_{\alpha+\delta_2} + \frac{\alpha_2+\alpha_3-\alpha_1-3}{2} I_{020}I_\alpha - \sum_{k=1}^{\alpha_1}\binom{\alpha_1}{k} I_{\alpha-k\delta_1+\delta_2}\hat\xi^{k,2},\\
&\mathrm{D}_y^{\mathrm{i}} I_\alpha = I_{\alpha+\delta_3} + \frac{\alpha_2+\alpha_3-\alpha_1-3}{2} I_{011}I_\alpha - \sum_{k=1}^{\alpha_1}\binom{\alpha_1}{k} I_{\alpha-k\delta_1+\delta_2}\hat\xi^{k,3}.
\end{aligned} \tag{2.22}$$

We only present the closed expressions for the first-order invariant derivatives of I_α with $|\alpha| \leqslant 3$:

$$\begin{aligned}
&\mathrm{D}_t^{\mathrm{i}} I_{110} = I_{210} - \tfrac32 I_{110}^2, \quad \mathrm{D}_x^{\mathrm{i}} I_{110} = I_{120} - \tfrac32 I_{110}I_{020} - I_{011}I_{020},\\
&\mathrm{D}_y^{\mathrm{i}} I_{110} = I_{111} - \tfrac32 I_{110}I_{011} - I_{020}I_{002},\\
&\mathrm{D}_t^{\mathrm{i}} I_{020} = I_{120} - \tfrac12 I_{110}I_{020}, \quad \mathrm{D}_x^{\mathrm{i}} I_{020} = I_{030} - \tfrac12 I_{020}^2, \quad \mathrm{D}_y^{\mathrm{i}} I_{020} = I_{021} - \tfrac12 I_{011}I_{020},\\
&\mathrm{D}_t^{\mathrm{i}} I_{011} = I_{111} - \tfrac12 I_{110}I_{011}, \quad \mathrm{D}_x^{\mathrm{i}} I_{011} = I_{021} - \tfrac12 I_{011}I_{020}, \quad \mathrm{D}_y^{\mathrm{i}} I_{011} = I_{012} - \tfrac12 I_{011}^2,\\
&\mathrm{D}_t^{\mathrm{i}} I_{002} = I_{102} - \tfrac12 I_{110}I_{002}, \quad \mathrm{D}_x^{\mathrm{i}} I_{002} = I_{012} - \tfrac12 I_{020}I_{002}, \quad \mathrm{D}_y^{\mathrm{i}} I_{002} = I_{003} - \tfrac12 I_{011}I_{002},\\
&\mathrm{D}_t^{\mathrm{i}} I_{210} = I_{310} - 2I_{110}I_{210}, \quad \mathrm{D}_x^{\mathrm{i}} I_{210} = I_{220} - 2I_{020}I_{210} - 2I_{011}I_{120} + (I_{011}^2 - I_{111})I_{020},\\
&\mathrm{D}_y^{\mathrm{i}} I_{210} = I_{211} - 2I_{011}I_{210} - 2I_{002}I_{120} + (I_{002}I_{011} - I_{102})I_{020},
\end{aligned}$$

$$
\begin{aligned}
&\mathrm{D}_t^{\mathrm{i}} I_{201} = I_{301} - 2I_{110}I_{201}, \quad \mathrm{D}_x^{\mathrm{i}} I_{201} = I_{211} - 2I_{020}I_{201} - 3I_{001}I_{111} + I_{011}^3,\\
&\mathrm{D}_y^{\mathrm{i}} I_{201} = I_{202} - 2I_{011}I_{201} - 2I_{002}I_{111} - I_{011}I_{102} + I_{002}I_{011}^2,\\
&\mathrm{D}_t^{\mathrm{i}} I_{120} = I_{220} - I_{110}I_{120}, \quad \mathrm{D}_x^{\mathrm{i}} I_{120} = I_{130} - I_{020}I_{120} - I_{011}I_{030},\\
&\mathrm{D}_y^{\mathrm{i}} I_{120} = I_{121} - I_{011}I_{120} - I_{002}I_{030},\\
&\mathrm{D}_t^{\mathrm{i}} I_{111} = I_{211} - I_{110}I_{111}, \quad \mathrm{D}_x^{\mathrm{i}} I_{111} = I_{121} - I_{020}I_{111} - I_{011}I_{021},\\
&\mathrm{D}_y^{\mathrm{i}} I_{111} = I_{112} - I_{011}I_{111} - I_{002}I_{021},\\
&\mathrm{D}_t^{\mathrm{i}} I_{102} = I_{202} - I_{110}I_{102}, \quad \mathrm{D}_x^{\mathrm{i}} I_{102} = I_{112} - I_{020}I_{102} - I_{011}I_{012},\\
&\mathrm{D}_y^{\mathrm{i}} I_{102} = I_{103} - I_{011}I_{102} - I_{002}I_{012},\\
&\mathrm{D}_t^{\mathrm{i}} I_{030} = I_{130}, \quad \mathrm{D}_x^{\mathrm{i}} I_{030} = I_{040}, \quad \mathrm{D}_y^{\mathrm{i}} I_{030} = I_{031},\\
&\mathrm{D}_t^{\mathrm{i}} I_{021} = I_{121}, \quad \mathrm{D}_x^{\mathrm{i}} I_{021} = I_{031}, \quad \mathrm{D}_y^{\mathrm{i}} I_{021} = I_{022},\\
&\mathrm{D}_t^{\mathrm{i}} I_{012} = I_{112}, \quad \mathrm{D}_x^{\mathrm{i}} I_{012} = I_{022}, \quad \mathrm{D}_y^{\mathrm{i}} I_{012} = I_{013},\\
&\mathrm{D}_t^{\mathrm{i}} I_{003} = I_{103}, \quad \mathrm{D}_x^{\mathrm{i}} I_{003} = I_{013}, \quad \mathrm{D}_y^{\mathrm{i}} I_{003} = I_{004}.
\end{aligned}
$$

In principle, it is possible to read off the generating differential invariants from the above split recurrence relations. The expressions for $I_{\alpha+\delta_1}$, $I_{\alpha+\delta_2}$ and $I_{\alpha+\delta_3}$ derived from (2.22) only involve first-order invariant derivatives of I_α and normalized invariants of orders not greater than $|\alpha|$. This implies that a generating set of differential invariants consists of invariantized derivatives which are minimal with respect to the usual partial ordering of derivatives and are not phantom invariants. We have four such minimal elements,

$$
I_{110} = \frac{\psi_{tx} - \psi_y\psi_{xx}}{\sqrt{|\psi_x|^3}}, \quad I_{020} = \frac{\psi_{xx}}{\sqrt{|\psi_x|}}, \quad I_{011} = \frac{\psi_{xy}}{\sqrt{|\psi_x|}}, \quad I_{002} = \frac{\psi_{yy}}{\sqrt{|\psi_x|}}.
$$

All the other invariantized derivatives are expressed via invariant derivatives of I_{110}, I_{020}, I_{011} and I_{002}. As was indicated above, not all differentiated differential invariants are necessarily functionally independent, which is encoded in syzygies of the algebra of differential invariants. Taking into account these syzygies can further reduce the number of generating differential invariants thereby allowing one a more concise description of the basis of differential invariants. In the present case, we find the following lower-order syzygies:

$$
\begin{aligned}
&\mathrm{D}_t^{\mathrm{i}} I_{011} - \mathrm{D}_y^{\mathrm{i}} I_{110} = I_{110}I_{011} + I_{020}I_{002},\\
&\mathrm{D}_t^{\mathrm{i}} I_{020} - \mathrm{D}_x^{\mathrm{i}} I_{110} = I_{020}(I_{110} + I_{011}),\\
&\mathrm{D}_y^{\mathrm{i}} I_{011} - \mathrm{D}_x^{\mathrm{i}} I_{002} = \tfrac{1}{2}I_{020}I_{002} - \tfrac{1}{2}I_{011}^2,\\
&\mathrm{D}_x^{\mathrm{i}} I_{011} - \mathrm{D}_y^{\mathrm{i}} I_{020} = 0,\\
&(\mathrm{D}_y^{\mathrm{i}})^2 I_{110} - \mathrm{D}_t^{\mathrm{i}}\mathrm{D}_x^{\mathrm{i}} I_{002} = \tfrac{1}{2}(\mathrm{D}_t^{\mathrm{i}} - I_{011})(I_{020}I_{002}) - (\mathrm{D}_y^{\mathrm{i}} + I_{011})(\tfrac{3}{2}I_{110}I_{011} + I_{020}I_{002})\\
&\qquad -I_{011}\mathrm{D}_y^{\mathrm{i}} I_{110} - I_{002}\mathrm{D}_y^{\mathrm{i}} I_{020},\\
&(\mathrm{D}_y^{\mathrm{i}})^2 I_{020} - (\mathrm{D}_x^{\mathrm{i}})^2 I_{002} = \tfrac{1}{2}\mathrm{D}_x^{\mathrm{i}}(I_{020}I_{002}) - \tfrac{1}{2}\mathrm{D}_y^{\mathrm{i}}(I_{011}I_{020}).
\end{aligned}
$$

From the two first syzygies we can express the invariants I_{011} and I_{002} via invariant derivatives of I_{110} and I_{020},

$$
\begin{aligned}
&I_{011} = \frac{\mathrm{D}_t^{\mathrm{i}} I_{020} - \mathrm{D}_x^{\mathrm{i}} I_{110}}{I_{020}} - I_{110},\\
&I_{002} = \frac{1}{I_{020}}(\mathrm{D}_t^{\mathrm{i}} - I_{110})\left(\frac{\mathrm{D}_t^{\mathrm{i}} I_{020} - \mathrm{D}_x^{\mathrm{i}} I_{110}}{I_{020}} - I_{110}\right) - \frac{\mathrm{D}_y^{\mathrm{i}} I_{110}}{I_{020}}.
\end{aligned}
$$

Another way of finding relations between generating invariants is to use the commutation relations between the operators of invariant differentiation. Evaluating each equality from (2.21) on an element I from the above generating set, we obtain a system of linear algebraic equations with respect to the other elements of these sets, which can be solved on the domain of the jet space where the determinant of the matrix associated with the system does not vanish. It is convenient to choose, e.g., $I = I_{020}$. Then, we derive the representations

$$
\begin{aligned}
I_{011} &= \frac{I_{020}\mathrm{D}^{\mathrm{i}}_y I_{020} - 2\varepsilon[\mathrm{D}^{\mathrm{i}}_x, \mathrm{D}^{\mathrm{i}}_y]I_{020}}{\mathrm{D}^{\mathrm{i}}_x I_{020}},\\
I_{110} &= \frac{2\varepsilon[\mathrm{D}^{\mathrm{i}}_t, \mathrm{D}^{\mathrm{i}}_x]I_{020} - I_{020}\mathrm{D}^{\mathrm{i}}_t I_{020}}{\mathrm{D}^{\mathrm{i}}_x I_{020}} - 2\varepsilon I_{011},\\
I_{002} &= \frac{[\mathrm{D}^{\mathrm{i}}_t, \mathrm{D}^{\mathrm{i}}_y]I_{020}}{\mathrm{D}^{\mathrm{i}}_x I_{020}} - \frac{\varepsilon}{2}\frac{\mathrm{D}^{\mathrm{i}}_t I_{020}}{\mathrm{D}^{\mathrm{i}}_x I_{020}} I_{011} - \frac{\varepsilon}{2}\frac{\mathrm{D}^{\mathrm{i}}_y I_{020}}{\mathrm{D}^{\mathrm{i}}_x I_{020}} I_{110},
\end{aligned}
$$

which are defined on the domain Ω_1 of the jet space where $\mathrm{D}^{\mathrm{i}}_x I_{020} \neq 0$, i.e., $\mathrm{D}^2_x(\sqrt{|\psi_x|}\,) \neq 0$.

As a result, it is straightforward to establish Theorem 2.2.

Bibliography

[1] Andreev V.K., Kaptsov O.V., Pukhnachov V.V. and Rodionov A.A., *Applications of group-theoretical methods in hydrodynamics*, Kluwer, Dordrecht, 1998.

[2] Andreev V.K. and Rodionov A.A., Group analysis of equations of planar flows of an ideal fluid in Lagrangian coordinates, *Dokl. Akad. Nauk SSSR* **298** (1988), 1358–1361.

[3] Arakawa A., Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. Part I, *J. Comp. Phys.* **1** (1966), 119–143.

[4] Batchelor G.K., Computation of the energy spectrum in homogeneous two-dimensional turbulence, *Phys. Fluids* **12**, Suppl. II (1969), 233–239.

[5] Benzi R., Paladin G., Patarnello S., Santangelo P. and Vulpiani A., Intermittency and coherent structures in two-dimensional turbulence, *J. Phys. A* **19** (1986), 3771–3784.

[6] Benzi R., Patarnello S. and Santangelo P., Self-similar coherent structures in two-dimensional decaying turbulence, *J. Phys. A* **21** (1988), 1221–1237.

[7] Bihlo A., Dos Santos Cardoso-Bihlo E.M. and Popovych R.O., Complete group classification of a class of nonlinear wave equations, *J. Math. Phys.* **53** (2012), 123515 (32 pages), arXiv:1106.4801.

[8] Bihlo A. and Popovych R.O., Lie symmetries and exact solutions of the barotropic vorticity equation, *J. Math. Phys.* **50** (2009), 123102 (12 pages), arXiv:0902.4099.

[9] Bihlo A. and Popovych R.O., Point symmetry group of the barotropic vorticity equation, in *Proceedings of 5th Workshop "Group Analysis of Differential Equations & Integrable Systems" (June 6–10, 2010, Protaras, Cyprus)*, 2011 pp. 15–27.

[10] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[11] Bihlo A. and Popovych R.O., Lie reduction and exact solutions of vorticity equation on rotating sphere, *Phys. Lett. A* **376** (2012), 1179–1184, arXiv:1112.3019.

[12] Bluman G. and Kumei S., *Symmetries and differential equations*, Springer, New York, 1989.

[13] Bracco A., McWilliams J.C., Murante G., Provenzale A. and Weiss J.B., Revisiting freely decaying two-dimensional turbulence at millennial resolution, *Phys. Fluids* **12** (2000), 2931–2941.

[14] Cheh J., Olver P.J. and Pohjanpelto J., Algorithms for differential invariants of symmetry groups of differential equations, *Found. Comput. Math.* **8** (2008), 501–532.

[15] Chen Q., Gunzburger M. and Ringler T., A scale-invariant formulation of the anticipated potential vorticity method, *Mon. Wea. Rev.* **139** (2011), 2614–2629.

[16] Danilov S. and Gurarie D., Forced two-dimensional turbulence in spectral and physical space, *Phys. Rev. E* **63** (2001), 061208 (12 pages).

[17] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[18] Dos Santos Cardoso-Bihlo E.M., Differential invariants for the Korteweg–de Vries equation, in *Proceedings of 6th Workshop "Group Analysis of Differential Equations & Integrable Systems" (June 17–21, 2012, Protaras, Cyprus)*, 2013 pp. 71–79.

[19] Dos Santos Cardoso-Bihlo E.M., Bihlo A. and Popovych R.O., Enhanced preliminary group classification of a class of generalized diffusion equations, *Commun. Nonlinear Sci. Numer. Simulat.* **16** (2011), 3622–3638, arXiv:1012.0297.

[20] Fels M. and Olver P.J., Moving coframes: I. A practical algorithm, *Acta Appl. Math.* **51** (1998), 161–213.

[21] Fels M. and Olver P.J., Moving coframes. II. Regularization and theoretical foundations, *Acta Appl. Math.* **55** (1999), 127–208.

[22] Fox S. and Davidson P., Freely decaying two-dimensional turbulence, *J. Fluid Mech.* **659** (2010), 351–364.

[23] Golovin S.V., Applications of the differential invariants of infinite dimensional groups in hydrodynamics, *Commun. Nonlinear Sci. Numer. Simul.* **9** (2004), 35–51.

[24] Huang H.P., Galperin B. and Sukoriansky S., Anisotropic spectra in two-dimensional turbulence on the surface of a rotating sphere, *Phys. Fluids* **13** (2001), 225–240.

[25] Ibragimov N.H., Aksenov A.V., Baikov V.A., Chugunov V.A., Gazizov R.K. and Meshkov A.G., *CRC handbook of Lie group analysis of differential equations. Vol. 2. Applications in engineering and physical sciences. Edited by N. H. Ibragimov*, CRC Press, Boca Raton, 1995.

[26] Kim P., Invariantization of numerical schemes using moving frames, *BIT Numerical Mathematics* **47** (2007), 525–546.

[27] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Phys. D* **237** (2008), 243–254.

[28] Kolmogoroff A.N., The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers **30** (1941), 299–303.

[29] Kraichnan R.H., Inertial ranges in two-dimensional turbulence, *Phys. Fluids* **10** (1967), 1417–1423.

[30] Legras B., Santangelo P. and Benzi R., High-resolution numerical experiments for forced two-dimensional turbulence, *Europhys. Lett.* **5** (1988), 37–42.

[31] Lilly D.K., Numerical simulation of developing and decaying two-dimensional turbulence, *J. Fluid Mech.* **45** (1971), 395–415.

[32] Maltrud M.E. and Vallis G.K., Energy spectra and coherent structures in forced two-dimensional and beta-plane turbulence, *J. Fluid Mech.* **228** (1991), 321–342.

[33] Oberlack M., *Invariant modeling in large-eddy simulation of turbulence*, *in: Annual research briefs*, Stanford University, 1997.

[34] Oberlack M., *Symmetrie, Invarianz und Selbstähnlichkeit in der Turbulenz*, Habilitationsschrift, Rheinisch–Westfählische Technische Hochschule Aachen, 2000.

[35] Oberlack M., A unified approach for symmetries in plane parallel turbulent shear flows, *J. Fluid Mech.* **427** (2001), 299–328.

[36] Olver P.J., *Application of Lie groups to differential equations*, Springer, New York, 2000.

[37] Olver P.J., Generating differential invariants, *J. Math. Anal. Appl.* **333** (2007), 450–471.

[38] Olver P.J. and Pohjanpelto J., Moving frames for Lie pseudo-groups, *Canadian J. Math.* **60** (2008), 1336–1386.

[39] Olver P.J. and Pohjanpelto J., Differential invariant algebras of Lie pseudo-groups, *Adv. Math.* **222** (2009), 1746–1792.

[40] Ovsiannikov L.V., *Group analysis of differential equations*, Acad. Press, New York, 1982.

[41] Pope S.B., *Turbulent flows*, Cambridge University Press, Cambridge, 2000.

[42] Popovych R.O. and Bihlo A., Symmetry preserving parameterization schemes, *J. Math. Phys.* **53** (2012), 073102 (36 pages), arXiv:1010.3010.

[43] Popovych R.O., Kunzinger M. and Eshraghi H., Admissible transformations and normalized classes of nonlinear Schrödinger equations, *Acta Appl. Math.* **109** (2010), 315–359.

[44] Razafindralandy D., Hamdouni A. and Béghein C., A class of subgrid-scale models preserving the symmetry group of Navier–Stokes equations, *Commun. Nonlinear Sci. Numer. Simul.* **12** (2007), 243–253.

[45] Rebelo R. and Valiquette F., Symmetry preserving numerical schemes for partial differential equations and their numerical tests, *J. Difference Equ. Appl.*, to appear, arXiv:1110.5921, 2012.

[46] Rhines P.B., Waves and turbulence on a beta-plane, *J. Fluid Mech.* **69** (1975), 417–443.

[47] Robinson W.A., On the self-maintenance of midlatitude jets, *J. Atmos. Sci.* **63** (2006), 2109–2122.

[48] Sadourny R. and Basdevant C., Parameterization of subgrid scale barotropic and baroclinic eddies in quasi-geostrophic models: Anticipated potential vorticity method, *J. Atmos. Sci.* **42** (1985), 1353–1363.

[49] Salmon R., *Lectures on geophysical fluid dynamics*, Oxford University Press, New York, 1998.

[50] Salmon R., A general method for conserving quantities related to potential vorticity in numerical models, *Nonlinearity* **18** (2005), R1–R16.

[51] Salmon R., A general method for conserving energy and potential enstrophy in shallow-water models, *J. Atmos. Sci.* **64** (2007), 515–531.

[52] Santangelo P., Benzi R. and Legras B., The generation of vortices in highresolution, twodimensional decaying turbulence and the influence of initial conditions on the breaking of selfsimilarity, *Phys. Fluids* **1** (1989), 1027–1034.

[53] Shen X. and Warhaft Z., The anisotropy of the small scale structure in high Reynolds number ($R_\lambda \sim 1000$) turbulent shear flow, *Phys. Fluids* **12** (2000), 2976–2989.

[54] Stensrud D.J., *Parameterization schemes: Keys to understanding numerical weather prediction models*, Cambridge University Press, Cambridge, 2007.

[55] Stull R.B., *An introduction to boundary layer meteorology*, vol. 13 of *Atmospheric Sciences Library*, Kluwer Academic Publishers, Dortrecht, 1988.

[56] Valiquette F. and Winternitz P., Discretization of partial differential equations preserving their physical symmetries, *J. Phys. A* **38** (2005), 9765–9783.

[57] Vallis G.K., *Atmospheric and Oceanic Fluid Dynamics*, Cambridge University Press, Cambridge, 2006.

[58] Vallis G.K. and Hua B.L., Eddy viscosity of the anticipated potential vorticity method, *J. Atmos. Sci.* **45** (1988), 617–627.

[59] Wilcox D.C., *Turbulence modeling for CFD*, DCW Industries, Inc., La Cañada, CA, 1993.

[60] Williams P.D., A proposed modification to the Robert–Asselin time filter, *Mon. Wea. Rev.* **137** (2009), 2538–2546.

Chapter 3

Conservative parameterization schemes

Alexander Bihlo[†] and George Bluman[‡]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*
Department of Mathematics and Statistics, McGill University, 805 Sherbrooke W., Montréal (QC) H3A 2K6, Canada

[‡] *Department of Mathematics, University of British Columbia*
Vancouver (BC) V6T 1Z2, Canada

E-mail*:* [†]*bihlo@crm.umontreal.ca,* [‡]*bluman@math.ubc.ca*

Parameterization (closure) schemes in numerical weather and climate prediction models account for the effects of physical processes that cannot be resolved explicitly by these models. Methods for finding physical parameterization schemes that preserve conservation laws of systems of differential equations are introduced. These methods rest on the possibility to regard the problem of finding conservative parameterization schemes as a conservation law classification problem for classes of differential equations. The relevant classification problems can be solved using the direct or inverse classification procedures. In the direct approach, one starts with a general functional form of the parameterization scheme. Specific forms are then found so that corresponding closed equations admit conservation laws. In the inverse approach, one seeks parameterization schemes that preserve one or more pre-selected conservation laws of the initial model. The physical interpretation of both classification approaches is discussed. Special attention is paid to the problem of finding parameterization schemes that preserve both conservation laws and symmetries. All methods are illustrated by finding conservative and invariant conservative parameterization schemes for systems of one-dimensional shallow-water equations.

3.1 Introduction

The problem of replacing the continuous governing equations of the atmosphere–ocean system by a discrete approximation is that in general no numerical scheme is capable of preserving all the geometrical features that the initial system of differential equations possesses. Among these features are symmetries and conservation laws. The lack of a numerical scheme in preserving

fundamental properties of the model has far-reaching consequences on the practical utility of the computed results. Simulating the earth system is a relevant but highly complex task and it involves an intricate interaction of theoretical insight, data handling and numerical modeling. Introducing errors in any of these tasks can lead to severe drifts of the forecasted state towards wrong attractors and thus to misleading weather and climate predictions. To remedy this challenge for the discretization part of the equations, several structure-preserving numerical integrators were developed [18, 28, 32, 47] that might eventually replace standard integration schemes.

Less research has been carried out so far on the slightly different but related problem of finding structure-preserving closure models or *parameterization schemes* for the subgrid-scale terms that inevitably arise when discretizing a nonlinear system of partial differential equations owing to the limited resolution one has to employ when integrating a numerical model. Attention to the importance of this issue was brought up in [20] and continued in [29], where the task of finding invariant subgrid-scale closure schemes for the filtered Navier–Stokes equations was investigated. Moreover, it was found in [21] that certain subgrid-scale closure models can also admit additional symmetries that have no counterparts in the original model to be closed. A subgrid-scale closure model that has such additional symmetries can indicate that the respective model is physically inadequate and underpin once more the preeminent role of symmetries in hydrodynamics, see also the discussion in [22].

The ideas formulated in [20] were recently also picked up in [6, 26] with the aim of formulating general algorithms for finding local parameterization schemes with prescribed invariance characteristics. These methods rely on the property that any generic parameterization ansatz, when introduced into an averaged system of differential equations turns this system into a class of closed differential equations. There exist powerful methods from the field of group analysis of differential equations [4, 4, 5, 5, 11, 14, 27, 28, 30, 42] that can be used to study the symmetry properties of such classes of differential equations in an algorithmic way. By a proper interpretation of these methods, one can turn them into effective tools for the construction and study of local parameterization schemes preserving invariance properties.

The fact that the local parameterization problem in general can be regarded as the study of the properties of classes of differential equations is also the key for the analog problem of finding parameterization schemes preserving conservation laws. Paralleling the task of finding parameterization schemes that preserve symmetry properties, this problem has important physical applications. Preserving physical conservation laws in the parameterization process is a natural requirement as there exist various processes that preserve e.g. energy or mass, but that cannot be resolved in a particular numerical model and thus have to be modeled in a simplified manner. When constructing closure models for such processes it is natural to require the closed system of equations to still preserve energy or mass as otherwise the physical consistency of the parameterization scheme is necessarily violated. Developing algorithmic methods that allow one to construct such parameterization schemes is thus a worthwhile endeavor.

A further point in favor of an extension of the tool kit of geometric closure schemes to include conservative parameterizations is related to the fact that symmetries and conservation laws of a system of differential equations $\mathcal{L}$ have different relations to the solutions of $\mathcal{L}$. By definition, a symmetry of $\mathcal{L}$ is a property of $\mathcal{L}$ itself without regard to posed initial and boundary conditions whereas a conservation law is a property holding for every solution of $\mathcal{L}$ irregardless of posed initial and boundary conditions. In particular, most of the nontrivial symmetries of equations in hydrodynamics are broken once classical boundary conditions are imposed (e.g. rigid walls or

even periodic domains). In contrast, the validity of a conservation law holds in the presence of all kinds of initial and boundary conditions.

Hence it is clear that it is physically relevant to study the relation between conservation laws, classes of differential equations and the parameterization problem. Shedding light on this relation is the purpose of the present paper. The main result of our study is a step towards the first systematic description of general methods for the construction of parameterization schemes that preserve or possess particular conservation laws.

This paper in organized as follows. In Section 3.2, the theory of conservative parameterization schemes is developed. Here, necessary terminology on conservation laws and group classification is introduced to demonstrate that the problem of finding conservative parameterization schemes can be regarded as a classification problem of conservation laws in classes of differential equations. This classification problem can be solved using both direct and inverse methods. These methods and the additional requirement on conservative parameterization schemes to be invariant with respect to a nontrivial symmetry can successfully be used to narrow down the vast possibility one generally has when constructing local subgrid-scale closure schemes for averaged or filtered differential equations. As an example, conservative and invariant conservative parameterization schemes are constructed for the system of one-dimensional shallow-water equations in Section 3.3. In Section 3.4, the results of the paper are summarized and suggestions are given for further research directions.

3.2 Conservative and invariant parameterization schemes

In this section we introduce some necessary terminology on symmetries and conservation laws, which are essential to formulate the theory of invariant and conservative parameterization schemes in a proper way. The exposition of the background material follows [1, 2, 5, 26–28, 42], to which we refer for a more thorough discussion of the underlying notions, methods and theoretical concepts.

3.2.1 General notions and statement of the problem

Let there be given a system of differential equations denoted by $\mathcal{L}$, which consists of L equations of the form $\Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots L$, where $x = (x^1, \dots, x^p)$ are the independent variables, $u = (u^1, \dots, u^q)$ are the dependent variables and $u^{(n)}$ denote all the derivatives of u with respect to x up to order n, with u being included as the zeroth order derivative.

Definition 3.1. A *local conservation law* of the system $\mathcal{L}$ is a divergence expression which vanishes on the solution set of the system $\mathcal{L}$ (denoted by $|_{\mathcal{L}}$),

$$\mathrm{D}_j\Phi^j|_{\mathcal{L}} = (\mathrm{D}_1\Phi^1 + \dots + \mathrm{D}_p\Phi^p)|_{\mathcal{L}} = 0. \tag{3.1}$$

The p-tuple of differential functions $\Phi = (\Phi^1(x, u^{(m)}), \dots, \Phi^p(x, u^{(m)}))$, $m \in \mathbb{N}_0$, is a *conserved current* of the associated conservation law.

Here and in the following, D_i is the operator of total differentiation with respect to x^i, defined by $\mathrm{D}_i = \partial_{x^i} + u^\alpha_{J,i}\partial_{u^\alpha_J}$, where $u^\alpha_J = \partial^{|J|}u^\alpha/\partial(x^1)^{j_1}\cdots\partial(x^p)^{j_p}$, $u^\alpha_{J,i} = \partial u^\alpha_J/\partial x^i$, $J = (j_1, \dots, j_p)$ is a multi-index, $j_i \in \mathbb{N}_0$ and $|J| = j_1 + \dots + j_p$. The summation convention over repeated indices is understood.

Definition 3.2. A local conservation law of the system $\mathcal{L}$ is *trivial* if the components of its conserved current Φ are of the form $\Phi^j = M^j(x, u^{(m)}) + H^j(x, u^{(m)})$, where the differential function M^j vanishes on the solution space of the system $\mathcal{L}$ and the tuple $(H^1, \dots, H^p)$ of differential functions is a null divergence, i.e. it satisfies $\mathrm{D}_j H^j = 0$ identically.

Trivial conservation laws satisfy the divergence condition in a trival way and thus contain no relevant information about the system $\mathcal{L}$. Consequently only nontrivial conservation laws are of interest below.

Definition 3.3. Two conserved currents Φ and Φ' represent the same conservation law (i.e. are *equivalent*) if their difference $\Phi - \Phi'$ is a conserved current associated with a trivial conservation law.

The above definition implies that conservation laws can only be found up to adding trivial conservation laws, i.e. there is not a single canonical representation of one and the same conservation law. Thus, formally the space of conservation laws can be defined as the set of elements from the factor space of the set of all conserved currents with respect to the subset of trivial conserved currents. See [28] for more details.

Conservation laws are conveniently found using the *multiplier approach*. This method rests on an equivalent recasting of the definition of a conservation law (3.1) in the form

$$\mathrm{D}_j \Phi^j(x, U^{(m)}) = \Lambda^l(x, U^{(r)}) \Delta_l(x, U^{(n)}), \tag{3.2}$$

where the differential functions $\Lambda = (\Lambda^1(x, U^{(r)}), \dots, \Lambda^L(x, U^{(r)}))$, $r \in \mathbb{N}_0$, are *conservation law* (CL) *multipliers*, also called the *characteristic of the conservation law associated with the conserved current* Φ. Note that expression (3.2) holds for arbitrary functions $U(x)$. It is then obvious that for solutions $U(x) = u(x)$ of the system $\mathcal{L}$ the right-hand side of the above expression vanishes and thus (3.2) reduces to the definition of a conservation law given above, provided that the characteristic Λ is non-singular on the solution manifold of $\mathcal{L}$.

Expression (3.2) can be converted into a system of determining equations for the multipliers Λ. This is facilitated by means of the Euler operator.

Definition 3.4. The *Euler operator* with respect to the dependent variable U^i is the differential operator given by

$$\mathsf{E}_i = \partial_{U^i} - \mathrm{D}_{j_1} \partial_{U^i_{j_1}} + \mathrm{D}_{j_1} \mathrm{D}_{j_2} \partial_{U^i_{j_1 j_2}} - \dots = (-\mathrm{D})^J \partial_{U^i_J}, \tag{3.3}$$

where $(-\mathrm{D})^J = (-\mathrm{D}_1)^{j_1} \cdots (-\mathrm{D}_p)^{j_p}$.

The importance of Euler operators in the study of local conservation laws lies in the property that they annihilate any divergence expression $\mathrm{D}_j \Phi^j$. In particular the CL multipliers $\Lambda = (\Lambda^1(x, U^{(r)}), \dots, \Lambda^L(x, U^{(r)}))$, $r \in \mathbb{N}_0$, yield a CL of $\mathcal{L}$ if and only of

$$\mathsf{E}_i(\Lambda^l \Delta_l) \equiv 0, \quad i = 1, \dots, q. \tag{3.4}$$

Equation (3.4) can be split with respect to Δ_l and its differential consequences. This yields an over-determined linear system of partial differential equations, which serve as the determining equations for the local CL multipliers of the system $\mathcal{L}$. Once these multipliers are found, the associated conserved currents Φ can be constructed using e.g. a *homotopy formula* [1, 2, 5].

We now move on to the precise statement of the *parameterization problem*. Given a system of differential equations $\mathcal{L}$, $\Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots, L$, consider a filtering operation

$$\mathcal{P}(u^i) = \bar{u}^i(x) = \int_\Omega u^i(y) G(y, x)\, \mathrm{d}y, \tag{3.5}$$

where $\mathrm{d}y = \mathrm{d}y^1 \cdots \mathrm{d}y^p$ and Ω is the domain of integration. Eq. (3.5) presents the convolution of the variable u^i with the filter kernel $G = G(y, x)$. The filter kernel $G(y, x)$ satisfies

$$\int_\Omega G(y, x)\, \mathrm{d}y = 1,$$

see [19, 20, 30]. This averaging operation can be used to decompose the instantaneous dependent variables u according to

$$u = \bar{u} + u'.$$

The average $\bar{u}$ is referred to as the resolved or grid-scale part of the dynamics, while u' includes the unresolved subgrid-scale fraction of u. As we do not mix different averaging methodologies in one and the same physical problem, we subsequently denote by a bar any mean value of u, irrespectively of what averaging operator is used in the concrete problem of interest.

Example 4. In the classical *Reynolds averaging* one uses the time average of u, which is defined by

$$\mathcal{P}_{\mathrm{R}}(u^i) = \bar{u}^i(x^*) = \lim_{T\to\infty} \frac{1}{T} \int_{t_0}^{t_0+T} u^i(t, x^*)\, \mathrm{d}t,$$

where t_0 denotes the initial time one starts to average. This time average follows from (3.5) upon setting $t = x^1$, $x^* = (x^2, \dots, x^p)$ and by factorizing

$$G(y, x) = G_1(y^1) \prod_{i=2}^{p} G_i(y^i - x^i) = \frac{\mathrm{H}_T(y^1)}{T} \prod_{i=2}^{p} \delta(y^i - x^i),$$

where H_T is the indicator function of the interval $[t_0, t_0 + T]$ and δ is the delta distribution. The time averaging is a Reynolds operator, i.e. it satisfies $\overline{\bar{u}_i u_j} = \bar{u}_i \bar{u}_j$. Owing to this property, in the splitting $u = \bar{u} + u'$ one has $\overline{u'} = 0$ since $\bar{\bar{u}} = \bar{u}$. The average over a product $u^i u^j$ thus gives $\overline{u^i u^j} = \bar{u}^i \bar{u}^j + \overline{u^{i'} u^{j'}}$, which is the classical Reynolds decomposition that introduces the Reynolds stresses $\overline{u^{i'} u^{j'}}$ into the averaged Navier–Stokes equations. In practical computation a finite $T < \infty$ has to be chosen and then $\bar{u} = \bar{u}(t_0, x^*)$, i.e. the mean value is still time-dependent.

Example 5. In *large eddy simulation* of turbulence, the classical Reynolds averaging as introduced in Example 4 is replaced by a spatial filtering approach defined by

$$\mathcal{P}_{\mathrm{LES}}(u^i) = \bar{u}^i(t, x^*) = \int_\Omega u^i(y) G(y, x) \mathrm{d}y,$$

for which the filter kernel in (3.5) is decomposed according to

$$G(y, x) = G_1(y^1 - t) \prod_{i=2}^{p} G_i(y^i - x^i) = \delta(y^1 - t) \prod_{i=2}^{p} G_i(y^i - x^i).$$

The filters defined this way are generally not Reynolds operators, i.e. now $\overline{u'} \neq 0$ as $\bar{\bar{u}} \neq \bar{u}$ and thus filtering over products $u^i u^j$ produces additional terms of forms not present in the Reynolds averaging approach, i.e. $\overline{u^i u^j} = \overline{\bar{u}^i \bar{u}^j} + \overline{u^{i'} \bar{u}^j} + \overline{\bar{u}^i u^{j'}} + \overline{u^{i'} u^{j'}}$.

With the aid of a particular averaging operator (3.5) the system $\mathcal{L}$ is converted into a system for the resolved part $\bar{u}$, which can be determined by measurements or in the course of a numerical simulation of the system $\mathcal{L}$. This is done by introducing the splitting $u = \bar{u}+u'$ into the system $\mathcal{L}$ followed by an application of a specific filtering (3.5), which leads to the averaged system of differential equations $\bar{\mathcal{L}}$ given by

$$\bar{\Delta}_l(x, \bar{u}^{(n)}, w) = 0, \quad l = 1, \dots, L. \tag{3.6}$$

In this expression the k-tuple $w = (w^1, \dots, w^k)$ includes all those terms that arise in the course of the averaging or filtering and cannot be determined from the knowledge of the mean or filtered values $\bar{u}^{(n)}$. For the Reynolds averaging introduced in Example 4, these are terms like $\overline{u^{i'}u^{j'}}$ (or higher order products as well as their derivatives), while in the case of the spatial filtering of Example 5, w would additionally include terms of the form $\overline{u^{i'}\bar{u}^j}$, etc. This is essentially the closure problem, i.e. Eqs. (3.6) include more unknown than known quantities. Thus, as they stand Eqs. (3.6) cannot be used for a numerical integration unless one expresses the additional unknowns w in terms of certain known expressions.

Definition 3.5. A *local* parameterization or subgrid-scale *closure* model assumes a functional relation between the unknown subgrid-scale terms w and the mean values $\bar{u}^{(r)}$, $r \in \mathbb{N}_0$, i.e.

$$w^i = f^i(x, \bar{u}^{(r)}), \quad i = 1, \dots, k, \tag{3.7}$$

for certain *parameterization functions* $f = (f^1, \dots, f^k)$.

Introducing a local parameterization scheme (3.7) into system (3.6) leads to a closed system of differential equations for the mean values $\bar{u}$. The inherent problem of this construction is that in most cases of interest the information contained in $\bar{u}$ and its derivatives is not sufficient to determine the entire subgrid-scale structure contained in w. The *art* of constructing physical parameterization schemes is to determine the parameterization functions f in such a manner that the assumption (3.7) will allow one to find $\bar{u}$ from the closed system $\bar{\mathcal{L}}$ with sufficient accuracy.

Finding suitable parameterization functions f that lead to realistic results for $\bar{u}$ can be rather tedious. One general methodology to restrict the vast number of possible forms for the parameterization functions is to choose them in such a manner that the resulting closed system of differential equations preserves certain nontrivial geometric properties such as conservation laws and/or symmetries. This motivates the following definition.

Definition 3.6. A local parameterization scheme is called *conservative* provided that the closed class of differential equations $\bar{\mathcal{L}}$ preserves certain nontrivial conservation laws of the initial system. A local parameterization scheme is called *invariant* provided that the closed class of differential equations $\bar{\mathcal{L}}$ preserves a nontrivial point symmetry group G of the initial system.

Conservative parameterization schemes can be found using techniques analogous to those for the *group classification* of classes of differential equations. A *class of differential equations* $\mathcal{L}|_{\mathcal{S}}$ is a system of differential equations of the form $\Delta_l(x, u^{(n)}, \theta(x, u^{(n)})) = 0$, $l = 1, \dots, L$, which is parameterized by a k-tuple $\theta = (\theta^1, \dots, \theta^k)$ of differential functions that satisfy a system of $K \in \mathbb{N}_0$ auxiliary differential equations of the form $S_j(x, u^{(n)}, \theta^{(m)}(x, u^{(n)})) = 0$, $j = 1, \dots, K$, the solution set of which is denoted by $\mathcal{S}$. The system of auxiliary equations in part specifies the properties of the class and it is regarded as a system for θ, i.e. x and $u^{(n)}$ play the role of

independent variables. To complete the description of the class $\mathcal{L}|_{\mathcal{S}}$ one usually takes into account a constitutive inequality, $\Sigma(x, u^{(n)}, \theta^{(m)}(x, u^{(n)})) \neq 0$, which guarantees that all equations from the class share some joint properties (e.g. a particular derivative does not vanish, all equations of $\mathcal{L}|_{\mathcal{S}}$ are linear or nonlinear, etc.).

It is the purpose of conservation law classification to systematically investigate the CLs of a class of differential equations. By substituting the general closure scheme (3.7) into the averaged system $\bar{\mathcal{L}}$ (3.6) one obtains a *class of closed differential equations* for $\bar{u}$,

$$\bar{\Delta}_l(x, \bar{u}^{(n)}, f(x, \bar{u}^{(r)})) = 0, \quad l = 1, \dots, L, \tag{3.8}$$

in which the parameterization functions f play the role of the arbitrary elements θ. In the following subsections we will introduce and discuss methods that allow one to specify the parameterization functions f in such a manner that the system (3.8) has certain nontrivial local CLs. The corresponding classification methods for finding parameterization schemes that possess nontrivial maximal Lie invariance groups where introduced in [11], see also [26]. Combinations of invariant and conservative parameterization schemes are also possible and will be discussed at the end of this section.

Remark 3.1. A problem related to the search for parameterization or closure schemes for the subgrid-scale terms arising in averaged differential equations is to search for *extensions* of differential equations that preserve some of the geometric features of the original differential equations $\mathcal{L}$. Physically, such extensions could be e.g. adding dissipation terms to a non-dissipative model or source or sink terms to transport equations. Depending on the nature of the included process, the addition of such extra terms may alter the structure of the initial system of differential equation but might still retain some of the geometric features of the original model. A possible research question is thus to construct extra terms for the system $\mathcal{L}\colon \Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots L$ in such a manner as to preserve certain CLs and/or symmetries of $\mathcal{L}$. Mathematically, this is done by investigating systems of the form $\Delta_l(x, u^{(n)}) = g_l(x, u^{(r)})$, for a certain L-tuple $g = (g^1(x, u^{(r)}), \dots g^L(x, u^{(r)}))$ of extensions. It is obvious that this system can be brought into the form (3.8) if the functions f are interpreted as the additional terms that extend the initial system $\mathcal{L}$ and no averaging operation is involved, i.e. $\bar{u}^{(n)} = u^{(n)}$. Consequently, the same methods as introduced below for solving the parameterization problem for system (3.8) can be used to tackle this kind of problem.

3.2.2 Conservative parameterizations via direct classification

In order to discuss the method for finding conservative parameterization schemes, the following definitions are useful.

Definition 3.7. An *equivalence transformation* φ from the class $\mathcal{L}|_{\mathcal{S}}$ is a point transformation on the space $(x, u^{(n)}, \theta)$, which is projectable on the spaces of $(x, u^{(n')})$, $0 \leq n' \leq n$, such that $\forall \theta \in \mathcal{S}\colon \theta' = \varphi\theta \in \mathcal{S}$ and the restriction of φ to the space of $(x, u^{(n)})$, denoted by $\varphi|_{(x,u^{(n)})}$, is a point transformation from $\mathcal{L}_\theta$ to $\mathcal{L}_{\theta'}$. Here, $\mathcal{L}_\theta$ and $\mathcal{L}_{\theta'}$ are equations from the class $\mathcal{L}|_{\mathcal{S}}$.

Thus, equivalence transformations are point transformations that map one system of differential equations from a given class $\mathcal{L}|_{\mathcal{S}}$ to another system of differential equations from the same class. The collection of all equivalence transformations forms a group, which is called the *equivalence group* $G^{\sim}$.

Definition 3.8. Let there be given two systems of differential equations from the class $\mathcal{L}|_{\mathcal{S}}$, denoted by $\mathcal{L}_\theta$ and $\mathcal{L}_{\theta'}$, which have CLs with conserved currents Φ and Φ', respectively. The pairs $(\mathcal{L}_\theta, \Phi)$ and $(\mathcal{L}_{\theta'}, \Phi')$ are *equivalent* with respect to the equivalence group $G^\sim$ if there exists a point transformation $\varphi \in G^\sim$ that transforms the system $\mathcal{L}_\theta$ to the system $\mathcal{L}_{\theta'}$ and which transforms the conserved current Φ in such a manner that $\tilde{\Phi} = \varphi(x, u^{(r)}, \Phi)$ and Φ' are equivalent conserved currents, see Definition 3.3.

In this definition, the action of a point transformation $\varphi \in G^\sim$ on a conserved current Φ has the explicit form

$$\tilde{\Phi}^i(\tilde{x}, \tilde{u}^{(r)}) = \frac{\mathrm{D}_{x^j}\tilde{x}^i}{|\mathrm{D}_x\tilde{x}|}\Phi^j(x, u^{(r)}), \quad i = 1, \dots, p,$$

where $|\mathrm{D}_x\tilde{x}|$ is the determinant of the matrix $(\mathrm{D}_{x^j}\tilde{x}^i)$. See [5, 9, 28] for more details.

The direct classification procedure for finding parameterization schemes with prescribed CLs can be formulated in the following way. For a given fixed general form of the parameterization functions f, determine those CLs that hold for any equation from the class (3.8) (i.e. for all admissible forms of f) and find all the inequivalent equations from that class that have additional CLs.

To make the classification problem tractable, one first chooses the general form of parameterization functions f one aims to study, i.e. one determines which variables x and $u^{(r)}$ the functions f should depend on. *This choice is physically motivated.* Once the general form of f (hence the system of auxiliary equations $\mathcal{S}$) is fixed, one can solve the classification problem taking into account the equivalence of CLs as embodied in Definition 3.8. This means that one determines the equivalence group of the general class of closed differential equations of interest and then solves the determining equations (3.4) for CL multipliers. One seeks those values of f (up to equivalence) for which the determining equations for CL multipliers yield additional multipliers beyond those for generic parameterization functions. An example for this procedure is given in Section 3.3.1.

The direct group classification method yields a list of inequivalent equations $\bar{\mathcal{L}}_f$ from the class $\bar{\mathcal{L}}|_{\mathcal{S}}$ that possess inequivalent nontrivial local CLs. Using this list of all possible conservative parameterization schemes from the predefined class, one can then test the different schemes obtained and select the most appropriate one as a candidate closure scheme for the process of interest that needs to be parameterized.

Physically, the method of finding conservative parameterization schemes using the direct classification approach might be most appropriate in the case when one seeks to represent processes that are not already included in the dynamics resulting from the system $\mathcal{L}$. The reason for this is that by means of the direct classification method one might obtain closed differential equations that have CLs not possessed by the original system $\mathcal{L}$.

3.2.3 Conservative parameterizations via inverse classification

A different ideology for finding parameterization schemes is the following. Assume that the original system of differential equations $\mathcal{L}$ has a certain number of nontrivial local CLs. The averaging of a differential equation certainly disturbs the geometric structure of the equation but it might nevertheless be desirable that the averaged system share some CLs of the original system of equations. An example for this is a process that conserves energy but needs to be parameterized in a given system of differential equations. For the sake of physical consistency,

the closed differential equations should conserve energy and thus only parameterization schemes that are compatible with energy conservation can be considered.

This discussion is related to what is called the *inverse classification problem* and in the framework of a conservative parameterization scheme, it can be formulated in the following way. Let there be given an initial system of differential equations $\mathcal{L}$. One first determines CLs holding for the original system of differential equations $\mathcal{L}$: $\Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots L$ through the CL multiplier approach. Depending on the complexity of the problem of interest, one might not be able to obtain an exhaustive description of all CLs but rather restricts oneself to CLs associated with characteristics $\Lambda(x, u^{(r)})$ for some fixed (often low-dimensional) r. Among the CLs of $\mathcal{L}$, one selects, using physical reasoning, the associated multipliers of those CLs that one aims to preserve in the course of the parameterization process. As in the case of the direct classification method, one next averages the system $\mathcal{L}$ and determines the general functional form of the parameterization functions f to be used in the class of parameterization schemes (3.7). The final step is to plug the class of averaged closed differential equations (3.8) into the determining equations (3.4) for CL multipliers. Since the multipliers that the resulting equations from the class (3.8) should admit are fixed, the determining equations for local CL multipliers are thus converted into a system of determining equations for the parameterization functions f. Solving this system leads to all equations from the class $\bar{\mathcal{L}}|_{\mathcal{S}}$ that have the same CL multipliers $\Lambda(x, \bar{u}^{(r)})$ as the original system $\mathcal{L}$, with $u^{(r)}$ being replaced by the corresponding averaged values $\bar{u}^{(r)}$.

A nontrivial question in this construction procedure is to determine in advance whether at least some systems from the class $\bar{\mathcal{L}}|_{\mathcal{S}}$ selected has the CLs associated with the chosen multipliers $\Lambda(x, u^{(r)})$ stemming from the original system $\mathcal{L}$, i.e. whether the determining equations (3.4) yield any nontrivial solutions. A natural strategy to overcome this problem of possibly triviality of the solution of (3.4) is to (i) either assume that the class of closed equations $\bar{\mathcal{L}}|_{\mathcal{S}}$ is rather wide (i.e. that the function f depends on a large number of variables from $\bar{u}^{(r)}$) or (ii) to only require a suitable small set of CLs being preserved by the parameterization scheme. From the physical point of view, the first strategy should be the method of choice.

Although this possible triviality of the solution of the determining equations for the parameterization functions and the associated failure in finding nontrivial conservative parameterization schemes seems undesirable, it nevertheless includes important physical evidence. It simply means that for the parameterization ansatz selected, no element of the class of closed equations can satisfy the requirement of retaining the desired CLs and thus might indicate that the initial parameterization ansatz was flawed.

Similar as the direct classification method, which might lead to conservative parameterization schemes that possess CLs which do not hold for the original system $\mathcal{L}$, conservative parameterization schemes derived using the inverse classification procedure may also yield additional CLs for the resulting closed system of differential equations. This should be investigated on a per case basis.

The inverse classification strategy might be best suited for processes already included in the full dynamics of the original system $\mathcal{L}$, but that cannot be explicitly resolved because of e.g. computational limitations.

Remark 3.2. The existence of at least the trivial solution of Eqs. (3.4) in the case when the CL multipliers are fixed and possible forms of f are sought, i.e. that the system of determining equations is compatible is guaranteed if (3.8) can be rewritten in the form

$$\Delta_l(x, \bar{u}^{(n)}) = g_l(x, \bar{u}^{(r)}), \quad l = 1, \dots, L,$$

where the left-hand side is the same as in the original system $\mathcal{L}$ provided that $\bar{u}^{(n)}$ is used in place of $u^{(n)}$ and the right-hand side is a functional combination of the parameterization functions f. Thus, if $g = 0$ (and hence $f = 0$), the above equation admits the same CL multipliers Λ as for the system $\mathcal{L}$ in which $\bar{u}^{(n)}$ is used instead of $u^{(n)}$.

3.2.4 Conservative and invariant parameterizations

Methods for finding parameterization schemes with symmetry properties using methods from the group analysis of differential equations were introduced in [6, 11, 20, 26]. There is neither a practical nor a theoretical objection against a parameterization scheme preserving both invariance *and* CL properties. Indeed, the compatibility of these two concepts was explicitly demonstrated by constructing CL and invariance preserving hyperdiffusion schemes for the two-dimensional barotropic vorticity equation on the beta-plane [6]. We now outline how to systematically construct *invariance and CL preserving parameterization schemes*.

As in the case of conservative parameterization schemes, two main methods are applicable to determine parameterization schemes with symmetry properties. These methods are straightforward applications of the group analysis of differential equations and the required techniques are based on the *direct* and the *inverse* approach to the symmetry classification problem, respectively. The key to the construction of conservative invariant parameterization schemes is that the closure models resulting from the conservative parameterization procedure as outlined in the previous two subsections are usually still classes of differential equations. These classes are generally narrower than the initial class given by Eqs. (3.8) but nevertheless include arbitrary constants or parameter functions with respect to which the usual symmetry classification problem can be carried out. We only outline the main ideas of this construction below as a more detailed exposition of the methods available in the field would require a substantial extension of the text. Moreover, the symmetry analysis and group classification of differential equations is a well-investigated subject. See [1–4, 4, 5, 5, 11, 14, 27, 28, 30, 38, 42] and references therein.

Special attention will be also paid to models that are derivable from *variational principles*, i.e. which are *Euler–Lagrange equations*. For such systems there is a close connection between symmetries and CLs that can be utilized to construct invariant conservative parameterization schemes. In particular, in this situation, all CL multipliers are symmetries but the converse is false [1, 2, 4, 5, 42].

Invariant conservative parameterizations via direct symmetry classification. In the direct symmetry classification approach one starts with a given class of differential equations and determines those symmetries that hold for all equations in the class. These symmetries form the *kernel* of maximal Lie invariance groups of equations in the class. The direct classification problem is solved by finding all equations in the class that have symmetry extensions with respect to the kernel and this investigation is carried out up to the equivalence imposed by the equivalence group $G^{\sim}$.

Depending on the complexity of the class of closed differential equations (3.8) with conservative properties, different strategies for solving the direct symmetry classification problem can be adopted. For simple classes depending only on a few arbitrary constants or parameter functions with few arguments, the *direct integration of the determining equations* of Lie symmetries up to equivalence using compatibility analysis is the method of choice. This method yields a complete list of inequivalent equations from classes of the form (3.8) that are both conservative and have nontrivial symmetry properties. If the structure of the class of (3.8) is too complicated for a

direct integration of the determining equations of Lie symmetries then the *algebraic method* of group classification can be used. With this method one aims to find symmetry extensions of the kernel that are induced by transformations from the equivalence group of the class under consideration. The algebraic method thus reduces the problem of finding symmetry extensions to the problem of finding inequivalent subgroups of the equivalence group $G^{\sim}$. Similar to the direct integration of the determining equations, the algebraic method of group classification can lead to the complete solution of the group classification problem, namely for classes of differential equations possessing the normalization property, see [27]. If the given class of equations is not normalized, then the algebraic method will not lead to a complete description of all possible inequivalent symmetry extensions of the kernel. One will still find those symmetry extensions that are induced by the equivalence transformations of the class of equations of interest but there can be other symmetry extensions of the kernel that cannot be found from the classification of the subgroups of the equivalence group. The algebraic method of direct group classification for classes that are not normalized is also known as *preliminary group classification*. A more detailed discussion of the techniques available in the field of direct group classification can be found in [5, 14, 27].

Irrespective of what method is used to (partially) solve the direct symmetry classification problem, all the systems of closed differential equations obtained in the classification procedure have the same CLs but different (inequivalent) maximal Lie invariance groups. The resulting closure schemes then have to be tested numerically to find the most suitable invariant and conservative representation for a given subgrid-scale process.

Invariant conservative parameterizations via inverse symmetry classification. The inverse symmetry classification problem is to find all those equations that have a prescribed symmetry property. See [11] for the first systematic outline of this problem for both ordinary and partial differential equations. Here, rather than starting from a given class of differential equations and describing the invariance properties of equations from this class as done in the direct symmetry classification, in the inverse classification one starts with a given Lie group of transformations and seeks to find the class of equations (up to some order n) that is invariant under the selected group. The inverse symmetry classification procedure rests on the following theorem [4, 5, 11, 38, 42]:

Theorem 3.1. *Let there be given a Lie group of transformations G acting on a manifold M. If the nth prolongation of G acts regularly on the nth order jet space J^n and if there exists a functionally independent system of nth order differential invariants $I_1, \dots, I_N$, $I_i = I_i(x, u^{(n)})$, then any nth order system of differential equations $\mathcal{L}$ admitting G as a symmetry group can be rewritten in terms of these differential invariants, i.e. $\Delta_l(x, u^{(n)}) = \tilde{\Delta}_l(I_1, \dots, I_N) = 0$, $l = 1, \dots, L$.*

This result is known as the *replacement theorem* [9, 10]. It implies the existence of a certain non-degenerate matrix $\Gamma^\kappa_l = \Gamma^\kappa_l(x, u^{(n)})$ such that the following holds for the system of differential equations $\mathcal{L}$: $\Delta_l(x, u^{(n)}) = \tilde{\Delta}_l(I_1, \dots, I_N) = 0$, $l = 1, \dots, L$,

$$\Gamma^\kappa_l(x, u^{(n)})\Delta_\kappa(x, u^{(n)}) = \tilde{\Delta}_l(I_1(x, u^{(n)}), \dots, I_N(x, u^{(n)})), \quad l = 1, \dots, L. \tag{3.9}$$

One can use the replacement theorem to construct parameterization schemes that have specified CL and invariance properties. To this end, one first determines the complete system of sth order differential invariants I_i of the maximal Lie invariance group G of the original unaveraged

system of differential equations $\mathcal{L}$ or of an appropriate subgroup $G^1 \subset G$, where $s = \min(n, r)$, n and r being the highest order derivatives arising in $\mathcal{L}$ and the parameterization ansatz (3.7), respectively. These invariants can be found either using infinitesimal techniques [4, 5, 11, 30, 42] or moving frames [9, 9, 10].

Secondly, once the invariants of G (or G^1) are known one can find the multipliers Γ_l^κ and thus obtain the invariant representation (3.9) of the system $\mathcal{L}$. Now suppose that the parameterization functions f in the class (3.8) have been determined in such a manner that the resulting equations from (3.8) have the desired CLs. As in Remark 3.2, one represents (3.8) in the solved form $\Delta_l(x, \bar{u}^{(n)}) = g_l(x, \bar{u}^{(r)})$, $l = 1, \dots, L$. These closed equations will be invariant under G (or G^1) for those functions g that satisfy the system of equations

$$\Gamma_l^\kappa(x, \bar{u}^{(n)}) g_\kappa(x, \bar{u}^{(r)}) = \tilde{g}_l(I_1, \dots, I_N), \quad l = 1, \dots, L, \tag{3.10}$$

using the same multipliers Γ_l^κ (replacing $u^{(n)}$ with $\bar{u}^{(n)}$), for some N-tuple of functions $\tilde{g}$ of the differential invariants I_i.

Similar to the discussion in Section 3.2.3, it is a nontrivial question to determine in advance whether condition (3.10) can be satisfied for a given set of conservative parameterization functions f and a chosen symmetry group G (or G^1). In general, the wider the symmetry group G (or G^1) is, the more specific are the forms of the differential invariants I_i and hence the more general the class of conservative parameterization schemes has to be in order to allow jointly for CL and invariance properties. On the other hand, if from the physical point of view it is required that the parameterization of a certain process should have specified invariance and CL properties, then failure in satisfying condition (3.10) may again point to an inappropriately chosen parameterization ansatz for f and thus is one more check for the consistency of a physical parameterization scheme.

Variational parameterizations for Lagrangian systems. As mentioned previously, there is a direct close connection between symmetries and CLs for systems of differential equations that can be derived from a variational principle. More precisely, for each one-parameter Lie group of point transformations (or, more generally, one-parameter group of higher order local transformations) that leaves invariant the functional

$$\mathcal{S}[u] = \int_\Omega L(x, u^{(n)}) \mathrm{d}x, \tag{3.11}$$

where $\mathrm{d}x = \mathrm{d}x_1 \cdots \mathrm{d}x_p$, to within a divergence, there is a local CL of the Euler–Lagrange equations associated with (3.11). This result is the celebrated *Noether theorem* [4, 5, 42]. Practically, the invariance of the functional $\mathcal{S}[u]$ under point transformations of the form $\tilde{x} = \tilde{x}(x, u, \varepsilon)$ and $\tilde{u} = \tilde{u}(x, u, \varepsilon)$, with associated infinitesimal generator $Q = \xi^i(x, u)\partial_{x^i} + \phi^\alpha(x, u)\partial_{u^\alpha}$, can be determined by checking whether the condition

$$Q^{(n)}(L) + L\mathrm{D}_i\xi^i = \mathrm{D}_i B^i \tag{3.12}$$

is satisfied, where $Q^{(n)}$ denotes the nth prolongation of Q and $B = (B^1(x, u^{(m)}), \dots, B^p(x, u^{(m)}))$ is some p-tuple of differential functions. In this case, it can be proved [4, 5, 42] that the characteristic $\eta = (\eta^1, \dots, \eta^q)$ of the vector field Q written in evolutionary form, which is given by $\eta^\alpha = \phi^\alpha - \xi^i u_i^\alpha$ is also a set of q local CL multipliers of the associated Euler–Lagrange equations of $\mathcal{S}[u]$, i.e. $\eta = \Lambda$. This means that a set of local conservation CL multipliers satisfies

the determining equations for local symmetries, in evolutionary form, of the corresponding Euler–Lagrange equations.

In practice, the Euler–Lagrange equations follow from taking the variational derivative of (3.11), which boils down to applying the Euler operator E_u to the Lagrangian function L. The following definitions and theorem will be useful for the construction of *variational parameterization schemes*, i.e. parameterization schemes that preserve the variational structure of a system of partial differential equations. A more comprehensive discussion on the relation between symmetries, CLs and variational forms can be found in [4, 5, 42].

Definition 3.9. Let there be given a system of differential equations $\mathcal{L}\colon \Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots, L$. The *linearizing operator* (*Fréchet derivative*) associated with $\mathcal{L}$ is the matrix-valued differential operator $\mathsf{D}_{\mathcal{L}}$ whose components are given by

$$(\mathsf{D}_{\mathcal{L}})_{\mu\nu} = \left(\frac{\partial \Delta_\mu}{\partial u^\nu} + \frac{\partial \Delta_\mu}{\partial u^\nu_{j_1}} \mathrm{D}_{j_1} + \cdots + \frac{\partial \Delta_\mu}{\partial u^\nu_{j_1 \cdots j_p}} \mathrm{D}_{j_1} \cdots \mathrm{D}_{j_p} \right) = \frac{\partial \Delta_\mu}{\partial u^\nu_J} \mathrm{D}_J.$$

Definition 3.10. The *adjoint* $\mathsf{D}^*_{\mathcal{L}}$ of the linearizing operator $\mathsf{D}_{\mathcal{L}}$ is the matrix-valued differential operator whose components are given by

$$(\mathsf{D}^*_{\mathcal{L}})_{\nu\mu} V^\mu = (-\mathrm{D})_J \left(\frac{\partial \Delta_\mu}{\partial u^\nu_J} V^\mu \right).$$

for any differential function $V = (V^1, \dots, V^L)$, where $V^l = V^l(x, u^{(n)})$. The operator $\mathsf{D}_{\mathcal{L}}$ is *self-adjoint* if and only if $\mathsf{D}^*_{\mathcal{L}} = \mathsf{D}_{\mathcal{L}}$.

Theorem 3.2. *Let $\mathcal{L}\colon \Delta_l(x, u^{(n)}) = 0$, $l = 1, \dots, L$ be a system of differential equations. The system $\mathcal{L}$ can be derived from a variational principle* (3.11) *if and only if the linearization operator $\mathsf{D}_{\mathcal{L}}$ associated with $\mathcal{L}$ is self-adjoint, i.e. $\mathsf{D}^*_{\mathcal{L}} = \mathsf{D}_{\mathcal{L}}$.*

Theorem 3.2 is the key for the construction of variational parameterization schemes. Suppose that the system $\mathcal{L}$ admits a variational form (3.11). Averaging of the equations from system $\mathcal{L}$ and assuming a general parameterization ansatz (3.7) leads to a system of equations that can be brought into the form

$$\Delta_l(x, \bar{u}^{(n)}) = g_l(x, \bar{u}^{(r)}), \quad l = 1, \dots, L. \tag{3.13}$$

See also Remark 3.2. Since the linearization operator associated with the left-hand side of the above expression is by supposition self-adjoint, the above closed system of differential equations (3.13) will remain variational if and only if

$$\mathsf{D}^*_g = \mathsf{D}_g, \tag{3.14}$$

i.e. the linearization operator associated with the right-hand sides of system (3.13) must be self-adjoint. This imposes the required conditions on the function $g = (g^1, \dots, g^L)$ to yield a variational parameterization scheme.

Since the one-dimensional shallow-water equations (3.17), which will be our running example for finding conservative parameterization schemes, are not derivable from a variational principle, we illustrate the idea of variational parameterization schemes for the potential Korteweg–de Vries equation.

Example 6. The potential Korteweg–de Vries (KdV) equation is obtained from the usual KdV equation $v_t + vv_x + v_{xxx} = 0$ by the differential substitution $v = u_x$, i.e.

$$\Delta_{\rm pKdV}(x, u^{(4)}) = u_{tx} + u_x u_{xx} + u_{xxxx} = 0. \tag{3.15}$$

This equation is variational with $L = \frac{1}{2}u_{xx}^2 - \frac{1}{6}u_x^3 - \frac{1}{2}u_t u_x$ being a Lagrangian. The maximal set of point symmetries of (3.15) is infinite dimensional and spanned by the vector fields

$$\partial_t, \quad \partial_x, \quad t\partial_x + x\partial_u, \quad \gamma(t)\partial_u, \quad 3t\partial_t + x\partial_x - u\partial_u,$$

where $\gamma = \gamma(t)$ runs through the set of smooth functions of t. The first four vector fields satisfy the variational symmetry condition (3.12) and are thus associated with local conservationl laws. The characteristics of these symmetries are (i) $\eta = -u_t$, (ii) $\eta = -u_x$, (iii) $\eta = x - tu_x$ and (iv) $\eta = \gamma(t)$ to which the following CLs $\Lambda(x, U, U_x, U_{xx}, U_{xxx})\Delta_{\rm pKdV}(x, U^{(4)}) = {\rm D}_t\rho + {\rm D}_x F$ correspond [5]:

Λ	ρ	F
$U_x^2 + 2U_{xxx}$	$\frac{1}{3}U_x^3 + U_x U_{xxx}$	$\frac{1}{4}U_x^4 + U_x^2 U_{xxx} - U_x U_{txx} + U_{xxx}^2 + U_{xx}U_{tx}$,
U_x	$\frac{1}{2}U_x^2$	$\frac{1}{3}U_x^3 + U_x U_{xxx} - \frac{1}{2}U_{xx}^2$,
$x - tU_x$	$\frac{1}{2}tU_x^2 + xU_x$	$-\frac{1}{3}tU_x^3 - tU_x U_{xxx} + \frac{1}{2}tU_{xx}^2 + \frac{1}{2}xU_x^2 - U_{xx} + xU_{xxx}$,
$\gamma(t)$	γU_x	$\frac{1}{2}\gamma U_x^2 + \gamma U_{xxx} - \gamma_t U$.

We average the potential KdV equation using Reynolds averaging to obtain $\bar{u}_{tx} + \bar{u}_x\bar{u}_{xx} + \bar{u}_{xxxx} = -\overline{u'_x u'_{xx}}$. To keep the computations as simple as possible we aim at finding the unclosed term $\overline{u'_x u'_{xx}}$ as a function of u, u_x and u_{xx} in such a manner that the closed equations from the class

$$u_{tx} + u_x u_{xx} + u_{xxxx} = g(u, u_x, u_{xx}),$$

are still derivable from a variational principle. From now on we omit bars over u to simplify the notation. The linearization operator associated with the right-hand side of this closed class of potential KdV equations is given by $\mathsf{D} = g_u + g_{u_x}{\rm D}_x + g_{u_{xx}}{\rm D}_x^2$, which has the adjoint $\mathsf{D}^* = g_{u_{xx}}{\rm D}_x^2 + (2{\rm D}_x g_{u_{xx}} - g_{u_x}){\rm D}_x + {\rm D}_x^2 g_{u_{xx}} - {\rm D}_x g_{u_x} + g_u$. Thus, condition (3.14) yields the system

$${\rm D}_x g_{u_{xx}} - g_{u_x} = 0, \quad {\rm D}_x^2 g_{u_{xx}} - {\rm D}_x g_{u_x} = 0$$

that must be satisfied by function g for an equation from the closed class of potential KdV equations to remain variational. Solving this system of differential equations yields $g = (c_2 u_x + c_1)u_{xx} + g^1(u)$ as a particular solution, where c_1 and c_2 are arbitrary constants and g^1 is an arbitrary function of u. Hence

$$u_{tx} + u_x u_{xx} + u_{xxxx} - (c_2 u_x + c_1)u_{xx} - g^1 = 0, \tag{3.16}$$

is the only admissible form of equations from the class of closed potential KdV equations that is an Euler–Lagrange equation. The associated Lagrangian of (3.16) is given by

$$L = \frac{1}{2}u_{xx}^2 - \frac{1}{6}(1 - c_2)u_x^3 - \frac{1}{2}u_t u_x - \frac{c_1}{2}u_x^2 - G(u),$$

where $G(u) = \int g^1 {\rm d}u$. For arbitrary g^1, the only point symmetries admitted by Eq. (3.16) are generated by the vector fields ∂_t and ∂_x, which are variational and give rise to two local CLs of (3.16).

3.3 Conservative closure schemes for the shallow-water equations

In this section we construct conservative parameterization schemes for the one-dimensional system of shallow-water equations. In nondimensional form, this system reads as

$$\begin{aligned}&\Delta_1(t,x,u^{(1)},h^{(1)}) = u_t + uu_x + h_x = 0,\\ &\Delta_2(t,x,u^{(1)},h^{(1)}) = h_t + uh_x + hu_x = 0,\end{aligned} \tag{3.17}$$

where u is the velocity and h is the height of the water column over a fixed reference level. For the sake of simplicity, we assume a flat bottom topography in which case h can be considered as the total height of the water column. The first equation is the momentum equation, the second equation the shallow-water continuity equation.

Reynolds-averaging the above system (3.17) with the averaging interval T being finite and following the averaging rule for products, $\overline{u^iu^j} = \bar{u}^i\bar{u}^j + \overline{u^{i'}u^{j'}}$, then leads to the averaged shallow-water equations

$$\begin{aligned}&\bar{u}_t + \bar{u}\bar{u}_x + \bar{h}_x = -\frac{1}{2}(\overline{u'^2})_x,\\ &\bar{h}_t + \bar{u}\bar{h}_x + \bar{h}\bar{u}_x = -(\overline{h'u'})_x.\end{aligned} \tag{3.18}$$

The right-hand sides of the above system are the subgrid-scale quantities that must be parameterized, i.e. it is necessary to find a functional relation that allows one to express these terms using only the grid-scale quantities, i.e.

$$-\frac{1}{2}(\overline{u'^2})_x = f(t,x,\bar{u}^{(r_1)},\bar{h}^{(q_1)}), \quad -(\overline{h'u'})_x = g(t,x,\bar{u}^{(r_2)},\bar{h}^{(q_2)}),$$

where $r_1, r_2, q_1, q_2 \in \mathbb{N}_0$. It is the purpose of this section to give two examples of forms of the parameterization functions f and g that lead to closed equations

$$u_t + uu_x + h_x = f, \quad h_t + uh_x + hu_x = g \tag{3.19}$$

possessing certain conservation laws. Here and in what follows, we omit the bars over averaged variables in the closed class of shallow-water equations since there is no risk of confusion as only averaged forms of dependent variables and their derivatives arise in such equations.

This notation is also convenient since in several of the examples given below, we consider the related problem of finding right-hand sides f and g in system (3.19) preserving certain CLs of system (3.17), i.e. we consider structure-preserving extensions of the shallow-water equations as discussed in Remark 3.1.

Before determining ansatzes for f and g that possess CLs or preserve certain CLs holding for the free shallow-water equations (3.18) it is instructive to determine the CLs of system (3.17). We restrict ourselves to multipliers that depend on t, x, U and H, i.e. $\Lambda^1 = \Lambda^1(t,x,U,H)$ and $\Lambda^2 = \Lambda^2(t,x,U,H)$, where U and H are arbitrary functions of t and x. The determining equations (3.4) for multipliers Λ^1 and Λ^2 in this case become

$$\begin{aligned}&\mathsf{E}_U(\Lambda^1\Delta_1(t,x,U^{(1)},H^{(1)}) + \Lambda^2\Delta_2(t,x,U^{(1)},H^{(1)})) \equiv 0,\\ &\mathsf{E}_H(\Lambda^1\Delta_1(t,x,U^{(1)},H^{(1)}) + \Lambda^2\Delta_2(t,x,U^{(1)},H^{(1)})) \equiv 0,\end{aligned} \tag{3.20}$$

where the Euler operators E_U and E_H are given by

$$\mathsf{E}_U = \partial_U - \mathrm{D}_t\partial_{U_t} - \mathrm{D}_x\partial_{U_x} + \cdots, \quad \mathsf{E}_H = \partial_H - \mathrm{D}_t\partial_{H_t} - \mathrm{D}_x\partial_{H_x} + \cdots.$$

Splitting the system (3.20) with respect to the derivatives of U and H, one obtains the following system of determining equations for CL multipliers

$$\Lambda^1_H - \Lambda^2_U = 0, \quad \Lambda^1_U - H\Lambda^2_H = 0, \quad \Lambda^2_t + U\Lambda^2_x + \Lambda^1_x = 0, \quad \Lambda^1_t + U\Lambda^1_x + H\Lambda^2_x = 0. \tag{3.21}$$

Differentiating the third equation in system (3.21) with respect to H and multiplying the resulting equation with H, one obtains, upon recombining with the first two equations, the equation

$$\Lambda^1_{Ut} + U\Lambda^1_{Ux} + H\Lambda^2_{Ux} = 0. \tag{3.22}$$

Differentiating the last equation in system (3.21) with respect to U and then combining the resulting equation with (3.22), one finds that $\Lambda^1_x = 0$. Then differentiating the second equation in (3.21) with respect to x one gets $\Lambda^2_{Hx} = 0$ and thus $\Lambda^2_{Ht} = 0$ in view of the third equation in system (3.21). Differentiation of the fourth equation in (3.21) with respect to x yields $\Lambda^2_{xx} = 0$ and hence $\Lambda^2_{tx} = 0$ due to the third equation.

Using these results, the integration of system (3.21) leads to

$$\begin{aligned} &\Lambda^1 = -c_1 tH + \lambda^1(U, H),\\ &\Lambda^2 = c_1(x - tU) + \lambda^2(U, H), \end{aligned} \tag{3.23}$$

where $c_1 = \text{const}$ and $\lambda^1 = \lambda^1(U, H)$ and $\lambda^2 = \lambda^2(U, H)$ are any functions satisfying the system $\lambda^1_H - \lambda^2_U = 0$ and $\lambda^1_U - H\lambda^2_H = 0$. Hence there are an infinite number of associated CLs, which reflects the possibility of linearizing the quasilinear system (3.17) using a hodograph transformation (interchanging the dependent and independent variables), see also Section 3.3.3. More details on the connection between CLs and linearization of partial differential equations can be found in [3, 5].

3.3.1 Conservative parameterization schemes via direct classification

In this subsection we give an example for the construction of conservative parameterization schemes using the technique of direct CL classification.

As an example, consider the problem of finding diffusion terms of the form

$$\begin{aligned} &u_t + uu_x + h_x - F(h, u_x, h_x)u_{xx} = 0,\\ &h_t + uh_x + hu_x = 0, \end{aligned} \tag{3.24}$$

allowing for CLs arising from multipliers of the form $\Lambda^1 = \Lambda^1(t, x, U, H)$ and $\Lambda^2 = \Lambda^2(t, x, U, H)$, where U and H are arbitrary functions of t and x. The problem is to first determine corresponding CLs arising for arbitrary F. Following this, we determine particular forms of F that yield additional CLs. The classification is done up to equivalence.

We should like to stress that this ansatz is a particular form of the more general class identified in Eq. (3.19) for which $f = F(h, u_x, h_x)u_{xx}$ and $g = 0$. We decided to work with this class subsequently because it is simple enough to allow us to illustrate the construction of conservative parameterization schemes in full detail without cluttering the presentation with technical details. On the other hand, this ansatz is physically interesting as it corresponds to adding a variable-coefficient viscous term to the free shallow-water equations.

Theorem 3.3. *The equivalence algebra $\mathfrak{g}^{\sim}$ of the class of one-dimensional dissipative shallow-water equations is generated by the following basis elements*

$$\partial_t, \quad \partial_x, \quad t\partial_x+\partial_u, \quad t\partial_t-u\partial_u-2h\partial_h-F\partial_F, \quad x\partial_x+u\partial_u+2h\partial_h+2F\partial_F. \tag{3.25}$$

Besides the associated continuous equivalence transformations (3.25)*, the class of equations* (3.24) *admits two independent discrete equivalence transformations, which are given by* $(t,x,u,h,F)\mapsto(-t,-x,u,h,-F)$ *and* $(t,x,u,h,F)\mapsto(-t,x,-u,h,-F)$*, respectively. The continuous and discrete equivalence transformations form the equivalence group $G^{\sim}$ of the class* (3.24)*.*

The determining equations for CL multipliers Λ^1 and Λ^2 are given by

$$\begin{aligned}
&\Lambda^1_H-\Lambda^2_U=0, \quad \Lambda^1_U-H\Lambda^2_H=0, \quad \Lambda^1_x+U\Lambda^2_x+\Lambda^2_t=0,\\
&\Lambda^1F_{H_x}=0, \quad H_x\Lambda^1F_{HU_x}+(U_x\Lambda^1_U+H_x\Lambda^1_H+\Lambda^1_x)F_{U_x}+2\Lambda^1_UF=0,\\
&2H_x\Lambda^1F_{HH_x}+2(U_x\Lambda^1_U+H_x\Lambda^1_H+\Lambda^1_x)F_{H_x}+\Lambda^1F_H+F\Lambda^1_H=0,\\
&H_x\Lambda^1F_{HH_x}+(\Lambda^1_x+U_x\Lambda^1_U+H_x\Lambda^1_H)F_{H_x}-\Lambda^1F_H-\Lambda^1_HF=0,\\
&H_x^2\Lambda^1F_{HH}+2(U_xH_x\Lambda^1_U+H_x^2\Lambda^1_H+H_x\Lambda^1_x)F_H+(U_x^2\Lambda^1_{UU}+2U_xH_x\Lambda^1_{UH}+\\
&H_x^2\Lambda^1_{HH}+2U_x\Lambda^1_{Ux}+2H_x\Lambda^1_{Hx}+\Lambda^1_{xx})F+U\Lambda^1_x+H\Lambda^2_x+\Lambda^1_t=0.
\end{aligned} \tag{3.26}$$

Assuming F to be arbitrary, one can split the above system with respect to the various derivatives of F, which then leads to the solution $\Lambda^1=0$ and $\Lambda^2=\text{const}$. *That is, for arbitrary F, the only CL admitted by system* (3.24)*, corresponding to the specified class of multipliers, is conservation of mass.*

We now consider particular forms of F for which system (3.24) possesses additional CLs. From the classifying condition $\Lambda^1F_{H_x}=0$ in system (3.26) it follows that $\Lambda^1=0$ when $F_{H_x}\neq0$. In this case, no CL extension exists. Thus, we only study the case of $F_{H_x}=0$ subsequently. In this case, the system of determining equations (3.26) simplifies significantly since all terms involving derivatives of F_{H_x} vanish. Consequently, it is possible to split the resulting equations with respect to the powers of H_x. The resulting system of determining equations is given by

$$\Lambda^1_H-\Lambda^2_U=0, \quad \Lambda^1_U-H\Lambda^2_H=0, \quad \Lambda^1_x+U\Lambda^2_x+\Lambda^2_t=0, \tag{3.27a}$$
$$\Lambda^1F_H+F\Lambda^1_H=0, \tag{3.27b}$$
$$(U_x\Lambda^1_U+\Lambda^1_x)F_{U_x}+2\Lambda^1_UF=0, \tag{3.27c}$$
$$(U_x^2\Lambda^1_{UU}+2U_x\Lambda^1_{Ux}+\Lambda^1_{xx})F+U\Lambda^1_x+H\Lambda^2_x+\Lambda^1_t=0. \tag{3.27d}$$

Equations (3.27a) do not involve the constitutive function F and thus can be integrated immediately. This results in

$$\begin{aligned}
\Lambda^1&=-\frac{1}{2}c_2tU^2+(c_1H+c_2x+c_3)U-c_2tH\ln H-\alpha_1(t)H-\frac{1}{2}\alpha_1''(t)x^2\\
&\quad-\alpha_2'(t)x+\alpha_3(t),\\
\Lambda^2&=\frac{1}{2}c_1U^2-(\alpha_1(t)+c_2t)\,U-(c_2(Ut-x)-c_3)\ln H+c_1H+\alpha_1'(t)x+\alpha_2(t),
\end{aligned} \tag{3.28}$$

where $\alpha_1(t)$, $\alpha_2(t)$ and $\alpha_3(t)$ are arbitrary smooth functions of t, a prime denotes the derivative with respect to t and c_i, $i=1,\dots,3$, are arbitrary constants.

Each of equations (3.27b)–(3.27d) explicitly involves the constitutive function F and these four equations are solved using compatibility analysis. Thus, different cases arise.

Case (I), $F_H = 0$. From Eq. (3.27b) it follows that $\Lambda^1_H = 0$ and Eq. (3.27a) implies that $\Lambda^2_U = 0$. From the form of the multipliers (3.28) we find that $c_1 = c_2 = 0$, $\alpha_1(t) = 0$ and thus the multipliers in this case are of the form

$$\Lambda^1 = c_3 U - \alpha_2'(t)x + \alpha_3(t), \quad \Lambda^2 = c_3 \ln H + \alpha_2(t),$$

Substituting this into Eq. (3.27d) results in

$$\alpha_2'(t)U + \alpha_2''(t)x - \alpha_3'(t) = 0,$$

which leads to $\alpha_2(t) = \text{const} = \alpha_2$ and $\alpha_3(t) = \text{const} = \alpha_3$. Eq. (3.27c) then reduces to

$$c_3(U_x F_{U_x} + 2F) = 0.$$

This equation implies that either (i) $c_3 = 0$ and $F = F(U_x)$ is arbitrary or (ii) $c_3 \neq 0$ and $F = c_0/U_x^2$, with c_0 being the integration constant. Note that in this case $U_x \neq 0$ must hold, which excludes uniform velocity fields from the following consideration. Since $c_0 \neq 0$ by assumption, we can use the transformations from the equivalence group $G^{\sim}$ to scale $c_0 = 1$. Thus in subcase (i) there is one additional CL which is of the form

$$\mathrm{D}_t u + \mathrm{D}_x \left(\frac{1}{2}u^2 + h - \int F(u_x)\mathrm{d}u_x \right) = 0. \tag{3.29}$$

In subcase (ii), the multipliers are

$$\Lambda^1 = c_3 U + \alpha_3, \quad \Lambda^2 = c_3 \ln H + \alpha_2$$

so that here there are two additional CLs. The CL associated with $\alpha_3 = \text{const}$ is CL (3.29) provided that $F = 1/u_x^2$. The second CL associated with c_3 is of the form

$$\mathrm{D}_t \left(\frac{1}{2}u^2 + (h \ln h - h) - t \right) + \mathrm{D}_x \left(\frac{1}{3}u^2 + h \ln h + \frac{1}{u_x} \right) = 0.$$

Case (II), $F_H \neq 0$. Substituting Λ^1 in the form given in Eq. (3.28) into Eq. (3.27b) we can split the resulting equation

$$\begin{aligned} &\Big(-\frac{1}{2}c_2 t U^2 + (c_1 H + c_2 x + c_3)U - c_2 t H \ln H - \alpha_1(t)H - \frac{1}{2}\alpha_1''(t)x^2 - \alpha_2'(t)x + \alpha_3(t) \Big) F_H \\ &+ (c_1 U - c_2 t(1 + \ln H) - \alpha_1(t))\, F = 0, \end{aligned}$$

with respect to powers of U and x since $F = F(H, U_x)$ only. Splitting with respect to U^2, x^2 and x implies that $c_2 = 0$, $\alpha_1(t) = \alpha_1^1 t + \alpha_1^0$ and $\alpha_2(t) = \text{const} = \alpha_2$, respectively, where $\alpha_1^1, \alpha_1^0 = \text{const}$. Differentiating the simplified equation twice with respect to t leads to $\alpha_3''(t) = 0$ or $\alpha_3(t) = \alpha_3^1 t + \alpha_3^0$, $\alpha_3^1, \alpha_3^0 = \text{const}$. The above equation thus simplifies to

$$\big((c_1 H + c_3)U - (\alpha_1^1 t + \alpha_1^0)H + \alpha_3^1 t + \alpha_3^0\big)\, F_H + \big(c_1 U - \alpha_1^1 t - \alpha_1^0\big)\, F = 0. \tag{3.30}$$

Plugging the simplified form of Λ^1 and Λ^2 into Eq. (3.27d) leads to $\alpha_3^1 = 0$. Splitting Eq. (3.30) with respect to t and U yields the following system of three equations

$$\alpha_1^1 H F_H + \alpha_1^1 F = 0, \quad (c_1 H + c_3)F_H + c_1 F = 0, \quad (\alpha_1^0 H - \alpha_3^0)F_H + \alpha_1^0 F = 0. \tag{3.31a}$$

At the same time, the remaining classifying equation (3.27c) gives

$$(c_1H + c_3)(2F + U_xF_{U_x}) = 0. \tag{3.31b}$$

System (3.31) allows one to find four inequivalent solutions. One either has (i) $(c_1, c_3, \alpha_1^1) = (0, 0, 0)$, (ii) $(c_1, c_3) \neq (0, 0)$, $\alpha_1^1 = 0$ and $\alpha_1^0 = \delta c_1$, $\alpha_3^0 = -\delta c_3$, (iii) $(c_1, c_3, \alpha_3^0) = (0, 0, 0)$ and $\alpha_1^0 = -\delta\alpha_1^1$ or (iv) $(c_3, \alpha_3^0) = (0, 0)$, $c_1 \neq 0$ and $\alpha_1^1 = \delta c_1$, $\alpha_1^0 = \varepsilon c_1$, $\delta, \epsilon = \text{const}$. In all other cases $F = 0$, which is excluded from consideration. In subcase (i), the classifiyng equations (3.31) are integrated to give

$$F = \frac{F^1(U_x)}{\alpha_1^0 H - \alpha_3^0},$$

where $F^1 = F^1(U_x)$ is an arbitrary non-vanishing smooth function of U_x and $\alpha_1^0 \neq 0$ since otherwise the assumption $F_H \neq 0$ would be contradicted. In this subcase, the multipliers Λ^1 and Λ^2 are given by

$$\Lambda^1 = \alpha_1^0 H - \alpha_3^0, \quad \Lambda^2 = \alpha_1^0 U + \alpha_2.$$

The two CLs that correspond to these characteristics are conservation of mass, which follows from the multipliers $(\Lambda^1, \Lambda^2) = (0, \alpha_2)$ and

$$\mathrm{D}_t(\alpha_1^0 hu - \alpha_3^0 u) + \mathrm{D}_x\left(\frac{1}{2}\alpha_1^0 h^2 - \frac{1}{2}\alpha_3^0 u^2 + \alpha_1^0 hu^2 - \alpha_3^0 h - \int g(u_x)\mathrm{d}u_x\right) = 0,$$

for the multipliers $(\Lambda^1, \Lambda^2) = (\alpha_1^0 H - \alpha_3^0, \alpha_1^0 U)$.

In subcase (ii), the solution of (3.31) is $F = c_0 U_x^{-2}/(c_1 H + c_3)$ and using the equivalence transformations from $G^{\sim}$ one can put $c_0 = 1$. The multipliers Λ^1 and Λ^2 are of the form

$$\Lambda^1 = (c_1 H + c_3)(U - \delta), \quad \Lambda^2 = \frac{1}{2}c_1 U^2 - \delta c_1 U + c_3 \ln H + c_1 H + \alpha_2.$$

The CLs associated with the found multipliers are again conservation of mass associated with $(\Lambda^1, \Lambda^2) = (0, \alpha_2)$ and

$$\mathrm{D}_t(c_1 hu + c_3 u) + \mathrm{D}_x\left(\frac{1}{2}(c_3 u^2 + c_1 h^2) + (c_1 u^2 + c_3)h + \frac{1}{u_x}\right) = 0$$

for the multiplier $(\Lambda^1, \Lambda^2) = (c_1 H + c_3, c_1 U)$. The third CL, corresponding to the multipliers $(\Lambda^1, \Lambda^2) = ((c_1 H + c_3)U, c_1 U^2/2 + c_3 \ln H + c_1 H)$, is

$$\mathrm{D}_t\left(\frac{1}{2}(c_1 h + c_3)u^2 + c_3(h \ln h - h) + \frac{1}{2}c_1 h^2 - t\right) + \mathrm{D}_x\left(\frac{1}{2}c_1 hu^3 + \frac{1}{3}c_3 u^3 + c_3 hu \ln h + c_1 h^2 u + \frac{u}{u_x}\right) = 0.$$

In subcase (iii) the solution of system (3.31) is $F = H^{-1}F^1(U_x)$, for an arbitrary non-vanishing smooth function $F^1(U_x)$. Here the multipliers are of the form

$$\Lambda^1 = \alpha_1^1(\delta - t)H, \quad \Lambda^2 = \alpha_1^1(x - Ut) + \delta a_1^1 U + \alpha_2.$$

The three CLs associated with these multipliers are again conservation of mass $(\Lambda^1 = 0, \Lambda^2 = \alpha_2)$, the CL

$$\mathrm{D}_t(xh - thu) + \mathrm{D}_x\left(xhu - thu^2 - \frac{1}{2}th^2 + t\int F^1(u_x)\mathrm{d}u_x\right) = 0,$$

corresponding to $(\Lambda^1,\Lambda^2)=(-tH,x-Ut)$ and the CL

$$\mathrm{D}_t(hu)+\mathrm{D}_x\left(\frac{1}{2}h^2+hu^2-\int F^1(u_x)\mathrm{d}u_x\right)=0$$

which stems from the multipliers $(\Lambda^1,\Lambda^2)=(H,U)$.

In the final subcase (iv) we obtain $F=c_0H^{-1}U_x^{-2}$ from the integration of the system (3.31). Again $c_0=1 \mod G^{\sim}$. The multipliers in this case are

$$\Lambda^1=(c_1(U-\delta t)-\epsilon c_1)H,\quad \Lambda^2=\frac{1}{2}c_1U^2+c_1H+\delta c_1(x-Ut)-\epsilon c_1U+\alpha_2.$$

Besides the obvious conservation of mass, the three other CLs include

$$\mathrm{D}_t(xh-thu)+\mathrm{D}_x\left(xhu-thu^2-\frac{1}{2}th^2-\frac{t}{u_x}\right)=0,$$

stemming from $(\Lambda^1,\Lambda^2)=(-tH,x-Ut)$,

$$\mathrm{D}_t(hu)+\mathrm{D}_x\left(\frac{1}{2}h^2+hu^2+\frac{1}{u_x}\right)=0$$

which is obtained from the multipliers $(\Lambda^1,\Lambda^2)=(H,U)$, and conservation of energy

$$\mathrm{D}_t\left(\frac{1}{2}hu^2+\frac{1}{2}h^2-t\right)+\mathrm{D}_x\left(\frac{1}{2}hu^3+h^2u+\frac{u}{u_x}\right)=0,$$

associated with the multipliers $(\Lambda^1,\Lambda^2)=(UH,U^2/2+H)$

Comparing the results of the two cases $F_H=0$ and $F_H\neq 0$ we have proved the following.

Theorem 3.4. *For $F\neq 0$ any equation from the class of dissipative systems of one-dimensional shallow-water equations* (3.24) *has at most four linearly independent conservation laws arising from multipliers of the form $\Lambda^1=\Lambda^1(t,x,U,H)$ and $\Lambda^2=\Lambda^2(t,x,U,H)$. A complete list of $G^{\sim}$-inequivalent equations and their associated conservation laws is given in Table 3.1, where $F^1=F^1(u_x)$ is an arbitrary non-vanishing smooth function of u_x and c_1 and c_2 are arbitrary constants with $c_2\neq 0$.*

3.3.2 Conservative parameterization schemes via inverse classification

In this subsection, two examples are presented that illustrate the procedure for finding conservative parameterization schemes through inverse group classification.

Parameterizations conserving energy, mass and momentum. In this example we focus on four physical CLs, namely conservation of mass-specific momentum, mass, momentum and energy which correspond to the multipliers

$$\Lambda^1=c_1+c_3H+c_4UH,\quad \Lambda^2=c_2+c_3U+c_4\left(\frac{1}{2}U^2+H\right), \tag{3.32}$$

for $c_1,\dots,c_4\in\mathbb{R}$ for the shallow-water equations (3.17). It can be checked that the above multipliers (3.32) satisfy the system of multiplier determining equations (3.21) and thus yield

Table 3.1: Conservation law classification of a class of one-dimensional dissipative shallow-water equations

Form of F	Conservation laws
$\forall F$	$\mathrm{CL}_1 = \mathrm{D}_t h + \mathrm{D}_x(hu) = 0$
$F = \dfrac{F^1(u_x)}{(c_1 h + c_2)}$	$\mathrm{CL}_1 = 0,$ $\mathrm{D}_t(c_1 hu + c_2 u) + \mathrm{D}_x\left(\frac{1}{2}c_1 h^2 + \frac{1}{2}c_2 u^2 + c_1 hu^2 + c_2 - \int F^1(u_x)\mathrm{d}u_x\right) = 0$
$F = \dfrac{1}{(c_1 h + c_2)u_x^2}$	$\mathrm{CL}_1 = 0,$ $\mathrm{D}_t(c_1 hu + c_2 u) + \mathrm{D}_x\left(\frac{1}{2}(c_2 u^2 + c_1 h^2) + (c_1 u^2 + c_2)h + \frac{1}{u_x}\right) = 0,$ $\mathrm{D}_t\left(\frac{1}{2}(c_1 h + c_2)u^2 + c_2(h\ln h - h) + \frac{1}{2}c_1 h^2 - t\right) +$ $\mathrm{D}_x\left(\frac{1}{2}c_1 hu^3 + \frac{1}{3}c_2 u^3 + c_2 hu\ln h + c_1 h^2 u + \frac{u}{u_x}\right) = 0$
$F = \dfrac{F^1(u_x)}{h}$	$\mathrm{CL}_1 = 0,$ $\mathrm{D}_t(xh - thu) + \mathrm{D}_x\left(xhu - thu^2 - \frac{1}{2}th^2 + t\int F^1(u_x)\mathrm{d}u_x\right) = 0,$ $\mathrm{D}_t(hu) + \mathrm{D}_x\left(\frac{1}{2}h^2 + hu^2 - \int F^1(u_x)\mathrm{d}u_x\right) = 0$
$F = \dfrac{1}{hu_x^2}$	$\mathrm{CL}_1 = 0,$ $\mathrm{D}_t(xh - thu) + \mathrm{D}_x\left(xhu - thu^2 - \frac{1}{2}th^2 - \frac{t}{u_x}\right) = 0,$ $\mathrm{D}_t(hu) + \mathrm{D}_x\left(\frac{1}{2}h^2 + hu^2 + \frac{1}{u_x}\right) = 0,$ $\mathrm{D}_t\left(\frac{1}{2}hu^2 + \frac{1}{2}h^2 - t\right) + \mathrm{D}_x\left(\frac{1}{2}hu^3 + h^2 u + \frac{u}{u_x}\right) = 0$

CLs for the shallow-water equations (3.17). The canonical forms $\mathrm{D}_t\rho + \mathrm{D}_x X = 0$ of the associated CLs are

c_1:	$\rho = u,$	$X = \frac{1}{2}u^2 + h$	mass-specific momentum,
c_2:	$\rho = h,$	$X = uh$	mass
c_3:	$\rho = uh,$	$X = u^2 h + \frac{1}{2}h^2$	momentum
c_4:	$\rho = \frac{1}{2}(u^2 h + h^2)$	$X = \frac{1}{2}hu^3 + h^2 u$	energy.

We now consider the problem of finding parameterization schemes (3.19) that preserve the above four multipliers. For the sake of demonstration we limit ourselves to constitutive functions of the form

$$f = f(x, u, h, u_x, h_x), \quad g = g(x, u, h, u_x, h_x),$$

where here and in what follows we omit the bars and the averaging of the dependent variables is to be understood. That is, we look for functions f and g that satisfy

$$\Lambda^1(\Delta_1 - f) + \Lambda^2(\Delta_2 - g) = \mathrm{D}_t\rho + \mathrm{D}_x X_0, \tag{3.33}$$

for the four multipliers (3.32) where $\Delta_1 = u_t + uu_x + h_x$, $\Delta_2 = h_t + uh_x + hu_x$ as before. Note that adding input terms to the shallow-water equations (3.17) will in general lead to modified

expressions for X, which is why we use X_0 in the CL (3.33). However, the conserved quantity ρ remains unchanged since the parameterization functions do not depend explicitly on t or derivatives of the unknown functions with respect to t.

Applying separately the Euler operators E_U and E_H with respect to U and H of Eq. (3.33), one obtains the system of determining equations for CL multipliers. Since the multipliers Λ^1 and Λ^2 are already prescribed, the resulting system

$$\mathsf{E}_U(\Lambda^1(\Delta_1 - f) + \Lambda^2(\Delta_2 - g)) = 0, \quad \mathsf{E}_H(\Lambda^1(\Delta_1 - f) + \Lambda^2(\Delta_2 - g)) = 0,$$

is now the system of the determining equations for the parameterization functions f and g. The determining equations can be split with respect to the constants c_1, c_2, c_3, c_4 and the unconstrained variables which are t, U_t, H_t, U_{tx}, H_{tx}, U_{xx} and H_{xx}. Splitting with respect to the highest derivatives arising yields the elementary equations

$$f_{U_xU_x} = f_{H_xU_x} = f_{H_xH_x} = g_{U_xU_x} = g_{H_xU_x} = g_{H_xH_x} = 0,$$

which can be integrated to give the following constrained form for the functions f and g,

$$\begin{aligned} f &= f^1(x,U,H)U_x + f^2(x,U,H)H_x + f^3(x,U,H),\\ g &= g^1(x,U,H)U_x + g^2(x,U,H)H_x + g^3(x,U,H). \end{aligned}$$

The remaining determining equations can be split to yield the system

$$\begin{aligned} &f^3 = g^3 = 0, \quad f^1_x = f^2_x = 0, \quad g^2 = f^1,\\ &g^1 - Hf^2 = 0, \quad f^1_H - f^2_U = 0, \quad f^1_U - Hf^2_H - f^2 = 0. \end{aligned} \tag{3.34}$$

System (3.34) can be integrated to give the most general form of the functions f and g admitting CLs of interest. In particular, one obtains

$$\begin{aligned} f &= f^1U_x + f^2H_x,\\ g &= f^2HU_x + f^1H_x. \end{aligned} \tag{3.35}$$

A particular class of solutions is given by

$$\begin{aligned} f^1 &= (\alpha_1 \sin\sqrt{b}U + \alpha_2\cos\sqrt{b}U)\left(\alpha_3 J_0(2\sqrt{bH}) + \alpha_4 Y_0(2\sqrt{bH})\right) + \alpha_5,\\ f^2 &= \frac{1}{\sqrt{H}}(\alpha_1\cos\sqrt{b}U - \alpha_2\sin\sqrt{b}U)\left(\alpha_3 J_1(2\sqrt{bH}) + \alpha_4 Y_1(2\sqrt{bH})\right). \end{aligned} \tag{3.36}$$

In this solution, $\alpha_1, \ldots, \alpha_5, b = \text{const}$, $b > 0$ and J_n and Y_n are the Bessel functions of the first and second kind, respectively.

The form (3.36) for f^1 and f^2 does not lead to a particularly physical parameterization ansatz. Physically more relevant forms for f and g can be found upon imposing other restrictions on these functions, which involves finding another interesting set of solutions of (3.34). An example of such a construction is the subclass of parameterization schemes of the form (3.35) which in addition to (3.34) satisfies the equation $f^1_U = 0$. In this case, the functions f^1 and f^2 in (3.35) are given by

$$f^1 = \beta_1 \ln H + \beta_2, \quad f^2 = \frac{\beta_1 U + \beta_3}{H},$$

where $\beta_1, \beta_2, \beta_3$ are arbitrary constants.

Conservative momentum dissipation schemes. As discussed in Remark 3.1, apart from the problem of finding conservative parameterization schemes another question of physical importance is to construct input terms that preserve some of the geometric structure of the initial system.

We illustrate this idea by constructing the most general dissipation term of the form

$$u_t + uu_x + h_x = f(x, h, h_x, u_x, u_{xx}), \quad h_t + uh_x + hu_x = 0,$$

for the shallow-water equations that preserves the multipliers $\Lambda^1 = c_0$ and $\Lambda^2 = c_1$, i.e. we set $f = f(x, h, h_x, u_x, u_{xx})$ and $g = 0$ in system (3.19). We do not aim at conserving momentum or energy in this case, because our aim in this example is to construct a dissipation for the shallow-water equations, which may violate energy and/or momentum conservation.

Using the same procedure as outlined in the previous example, we find that f should be of the form

$$f = f^1 u_{xx} + \left(\int f^1_h \mathrm{d}u_x + f^2 \right) h_x + \int \left(\int f^1_{xh} \mathrm{d}u_x + f^2_x \right) \mathrm{d}h + cu_x + f^3, \tag{3.37}$$

where $c \in \mathbb{R}$, $f^1 = f^1(x, h, u_x)$, $f^2 = f^2(x, h)$ and $f^3 = f^3(x)$.

From the form of (3.37), one observes that the requirement of conserving both mass and mass-specific momentum leads to a quasi-linear dissipation scheme, i.e. f is linear in terms of u_{xx}.

3.3.3 Conservative and invariant parameterization

We now turn to the problem of finding parameterization schemes that are both conservative and preserve certain symmetries of the original (unaveraged) system of differential equations. We illustrate this idea with the two examples of the previous section.

The maximal Lie invariance algebra $\mathfrak{g}$ of the system of one-dimensional shallow-water equations (3.17) is infinite dimensional and has the following basis elements

$$\begin{aligned} &t\partial_t + x\partial_x, \quad x\partial_x + u\partial_u + 2h\partial_h, \quad t\partial_x + \partial_u, \\ &(2x - 6tu)\partial_t + (6h - 3u^2)t\partial_x + (u^2 + 4h)\partial_u + 4hu\partial_h, \quad \tau(h, u)\partial_t + \zeta(h, u)\partial_x, \end{aligned} \tag{3.38}$$

where the functions τ and ζ run through the set of solutions of the system

$$\zeta_h - u\tau_h + \tau_u = 0, \quad \zeta_u - u\tau_u + h\tau_h = 0.$$

The infinite dimensional part of $\mathfrak{g}$ indicates the existence of a linearization transformation, which is given through the hodograph transformation, i.e. $\tau = t$ and $\zeta = x$ are the new dependent variables and u and h are the new independent variables.

The question of which symmetries one aims to preserve when constructing a (conservative) parameterization scheme should be answered using physical arguments. For example, processes that are to be parameterized in the framework of classical mechanics should be represented in such a manner so as to be invariant with respect to the Galilean group. Choices for subgroups to be preserved by a parameterization scheme can be also motivated from compatibility with certain boundary-value problems, see also the related discussion in [4].

As outlined in Section 3.3.3, when constructing parameterization schemes that are required to be both invariant and conservative one can follow two ways, namely using direct or inverse

group classification. We now use the examples worked out in the previous section to illustrate both ways.

Invariant conservative parameterization using direct classification. The direct group classification method can be illustrated with the first example from the previous section. Essentially, the only freedom one has left with the parameterization (3.35)–(3.36) is to set to zero some of the constants $\alpha_1, \dots, \alpha_5$ and investigate which symmetries the resulting systems have. This analysis should be done up to equivalence of the class of equations of the form (3.35)–(3.36).

The kernel of the maximal Lie invariance algebra from this class is given by the subalgebra of $\mathfrak{g}$ consisting of the basis elements

$$\mathfrak{g}^{\cap} = \langle t\partial_t + x\partial_x, \partial_t, \partial_x \rangle.$$

It turns out that the only extension of this kernel algebra arises when $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0$ and $\alpha_5 = c \neq 0$. However, in this case the transformation $\tilde{u} = u - c$ maps the resulting case to the initial shallow-water equations (3.17), i.e. the parameterization becomes trivial. Stated in another way, the constant α_5 is inessential for the classification problem of the class (3.35)–(3.36), which has the two inequivalent solutions $(\alpha_1, \alpha_2, \alpha_3, \alpha_4) \neq (0, 0, 0, 0)$ and $(\alpha_1, \alpha_2, \alpha_3, \alpha_4) = (0, 0, 0, 0)$. Only the first solution leads to a nontrivial parameterization scheme. However, as in this case the admitted symmetry algebra is only $\mathfrak{g}^{\cap}$ and thus does not include Galilean transformations, one might question the physical relevance of the resulting parameterization especially in the light of the discussion in the beginning of this section.

A similar analysis could be carried out with the other parameterization schemes of the form (3.35) that can be constructed by finding only particular solutions of the system (3.34). As in the previous case, these classes of parameterizations essentially only depend on certain constants. The classification problem then basically reduces to finding those constants that can be set to zero by a proper transformation of the equation variables (inessential constants) and studying the classification problem with respect to the remaining (essential) constants. This is a straightforward task and is not considered further in this paper.

Invariant conservative parameterization using inverse classification. We now focus on the problem of finding elements of the class of shallow-water equations (3.19) with f given by (3.37) and $g = 0$ that are invariant under a certain subgroup G^1 of the maximal Lie invariance (pseudo)group G of the shallow-water equations. That is, we use the inverse symmetry classification strategy to find models from this class. As a symmetry subgroup G^1, we single out the four-parameter subgroup of G that is generated by the four-dimensional Lie subalgebra $\mathfrak{g}^1$ of $\mathfrak{g}$ with the basis elements

$$\partial_t, \quad \partial_x, \quad t\partial_x + \partial_u, \quad t\partial_t + (1+d)x\partial_x + du\partial_u + 2dh\partial_h,$$

for an arbitrary constant $d \in \mathbb{R}$. The reason for choosing this particular subalgebra is that for physical arguments we require our dissipation scheme to be invariant under the Galilean group (generated by the first three elements of $\mathfrak{g}^1$) and to have a scaling symmetry. We only require invariance under a single scaling instead of the two scalings admitted by the original shallow-water equations (3.17) as adding dissipation to a hydrodynamical system usually breaks one scaling symmetry. For example, the shallow-water system with classical linear dissipation, $f = u_{xx}$ has the above scaling symmetry provided $d = -1/2$.

As discussed in Section 3.3.3 one can use the replacement theorem to construct parameterization schemes that are both conservative and invariant. To this end, we determine the second

order differential invariants of the Lie algebra $\mathfrak{g}^1$. They can be found by using infinitesimal techniques through prolongation of $\mathfrak{g}^1$ to the action on second derivatives of u and h, and then invoking the infinitesimal invariance criterion [4, 5, 11, 30, 42]. These invariants can also be found using the moving frame method [9, 9, 10]. In particular, on the space of equation variables for the class (3.19) with f satisfying (3.37) and $g = 0$, which is the subspace of the second jet space J^2 with coordinates $(t, x, u, h, u_t, u_x, h_t, h_x, u_{xx})$, there are five elementary invariants given by

$$\begin{aligned} &I_1 = hu_x^{2d}, \quad I_2 = h_x u_x^{d-1}, \quad I_3 = u_{xx}u_x^{-(2+d)}, \\ &I_4 = u_x^{d-1}(u_t + uu_x), \quad I_5 = u_x^{2d-1}(h_t + uh_x). \end{aligned} \tag{3.39}$$

The invariant representation of system (3.17) is therefore

$$I_4 + I_2 = u_x^{d-1}\Delta_1, \quad I_5 + I_1 = u_x^{2d-1}\Delta_2$$

from which it follows that the multipliers Γ_1^1, Γ_1^2, Γ_2^1 and Γ_2^2 in system (3.9) are $\Gamma_1^1 = u_x^{d-1}$, $\Gamma_1^2 = 0$, $\Gamma_2^1 = 0$ and $\Gamma_2^2 = u_x^{2d-1}$.

The system of shallow-water equations with the dissipation scheme (3.37) is invariant under the group generated by the elements of $\mathfrak{g}^1$ provided that

$$\Gamma_1^1 f = \tilde{f}(I_1, I_2, I_3) \tag{3.40}$$

holds for some function $\tilde{f}$ that can depend at most on the invariants I_1, I_2 and I_3 since none of the functions f^1, f^2 and f^3 in (3.37) depends on u_t or h_t.

We now determine some dissipation schemes that fulfill the above requirement. For the sake of simplicity, we assume that $f^2 = f^3 = 0$, since we are mainly interested in the second-order term proportional to u_{xx} and thus in finding functions f^1 that lead to invariant and conservative diffusion schemes. As none of the invariants I_1, I_2 and I_3 depends on x we have that $f_x^1 = 0$. The function $\tilde{f}$ should depend linearly on I_3 to match with u_{xx} in f. Comparing the coefficients of u_{xx} we find that

$$f^1 = u_x^{-(1+2d)}\alpha(I_1).$$

Note that α cannot depend on I_2, since f^1 does not depend on h_x. The remaining condition that has to hold is that

$$h_x \int f_h^1 \mathrm{d}u_x = u_x^{1-d}\beta(I_1, I_2).$$

The function β now cannot depend on I_3 as there is no u_{xx} term in the above left-hand side. Since the left-hand side is linear in h_x, we find that $\beta = I_2\gamma(I_1)$ and thus get

$$\int \alpha_{I_1} u_x^{-1} \mathrm{d}u_x = \gamma(I_1).$$

which imposes a relation between the functions α and γ. This relation is particularly straightforward to evaluate for polynomial functions α. To give an example, let us set $\alpha = 2cdI_1^2$, for $c \in \mathbb{R}$. This leads to the shallow-water equations with dissipation in the form

$$u_t + uu_x + h_x = \mathrm{D}_x\left(ch^2(u_x)^{2d}\right), \quad h_t + uh_x + hu_x = 0.$$

The usual linear dissipation $f = \nu u_{xx}$, $\nu \in \mathbb{R}$, falls into this class when putting $d = -1/2$ and using $\alpha(I_1) = \nu$.

3.4 Conclusion

In this paper we have studied the problem of finding physical parameterization schemes that lead to closed systems of averaged differential equations which possess nontrivial local conservation laws. A main motivation for our work is that by its formulation, one cannot expected to find an exact solution to the parameterization problem. The determination of the entire subgrid-scale structure of a real-world process when one has at their disposal only the grid-scale information is not feasible for nontrivial physical problems. Hence, any auxiliary information that can be used to limit the possible form of a parameterization scheme by imposing some physically and geometrically relevant structural constraints is highly useful. The preservation of symmetries and conservation laws can serve as such relevant constraints since these two properties are closely linked with the physics encoded in a system of differential equations.

A systematic toolbox of methods that allows one to systematically find parameterization schemes with symmetry properties using group classification techniques was formulated in [6, 26]. As far as we know, this paper is the first to use the analog toolbox for finding parameterization schemes preserving conservation laws.

The results of this paper illustrate that the requirement of preserving a particular set of conservation laws when constructing physical parameterization schemes can lead to rather specific forms for these schemes. This is in striking contrast to the case of invariant parameterization schemes, since here there are in general an infinite number of possibilities to construct a subgrid-scale closure possessing a prescribed maximal Lie invariance group. Moreover, as to be expected, the more conservation laws one aims to conserve when constructing a subgrid-scale closure or any other additional model for a system of differential equations, the less freedom one has to adjust the closure by including other desirable properties. Thus, the requirement of preserving certain conservation laws (and symmetries) can lead to rather specific parameterization ansatzes which in consequence could potentially simplify the construction and testing procedures for subgrid-scale closure models.

In the present paper, the primary focus of the presentation was to give a careful exposition of the different ideologies for finding conservative closure schemes. The system of one-dimensional shallow-water equations served as a proof-of-the-concept example but did not reveal new physical insights. More realistic examples of conservative parameterization schemes for the governing equations of hydro-thermodynamics will be presented elsewhere.

Acknowledgements

For deriving the determining equations of characteristics of conservation laws we have used the package `GEM` by Alexei Cheviakov [13]. This research was supported by the Austrian Science Fund (FWF), project J3182–N13 (AB) and the National Sciences and Engineering Research Council of Canada (GB).

Bibliography

[1] Anco S. and Bluman G., Direct construction method for conservation laws of partial differential equations. Part I: Examples of conservation law classifications, *Eur. J. App. Math.* **13** (2002), 545–566.

[2] Anco S. and Bluman G., Direct construction method for conservation laws of partial differential equations. Part II: General treatment, *Eur. J. App. Math.* **13** (2002), 567–585.

[3] Anco S., Bluman G. and Wolf T., Invertible mappings of nonlinear PDEs to linear PDEs through admitted conservation laws, *Acta Appl. Math.* **101** (2008), 21–38.

[4] Basarab-Horwath P., Lahno V. and Zhdanov R., The structure of Lie algebras and the classification problem for partial differential equations, *Acta Appl. Math.* **69** (2001), 43–94.

[5] Bihlo A., Dos Santos Cardoso-Bihlo E.M. and Popovych R.O., Complete group classification of a class of nonlinear wave equations, *J. Math. Phys.* **53** (2012), 123515 (32 pages), arXiv:1106.4801.

[6] Bihlo A., Dos Santos Cardoso-Bihlo E.M. and Popovych R.O., Invariant parameterization and turbulence modeling on the beta-plane, arXiv:1112.1917, 25 pp., 2012.

[7] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[8] Bluman G. and Kumei S., *Symmetries and differential equations*, Springer, New York, 1989.

[9] Bluman G., Temuerchaolu and Anco S.C., New conservation laws obtained directly from symmetry action on a known conservation law, *J. Math. Anal. Appl.* **322** (2006), 233–250.

[10] Bluman G.W., Cheviakov A.F. and Anco S.C., *Application of symmetry methods to partial differential equations*, Springer, New York, 2010.

[11] Bluman G.W. and Cole J.D., *Similarity methods for differential equations*, Springer, New York, 1974.

[12] Cheh J., Olver P.J. and Pohjanpelto J., Algorithms for differential invariants of symmetry groups of differential equations, *Found. Comput. Math.* **8** (2008), 501–532.

[13] Cheviakov A.F., GeM software package for computation of symmetries and conservation laws of differential equations, *Comput. Phys. Comm.* **176** (2007), 48–61.

[14] Dos Santos Cardoso-Bihlo E.M., Bihlo A. and Popovych R.O., Enhanced preliminary group classification of a class of generalized diffusion equations, *Commun. Nonlinear Sci. Numer. Simulat.* **16** (2011), 3622–3638, arXiv:1012.0297.

[15] Fels M. and Olver P.J., Moving coframes: I. A practical algorithm, *Acta Appl. Math.* **51** (1998), 161–213.

[16] Fels M. and Olver P.J., Moving coframes. II. Regularization and theoretical foundations, *Acta Appl. Math.* **55** (1999), 127–208.

[17] Frank J. and Reich S., The Hamiltonian particle–mesh method for the spherical shallow water equations, *Atmos. Sci. Lett.* **5** (2004), 89–95.

[18] Gassmann A. and Herzog H.J., Towards a consistent numerical compressible non-hydrostatic model using generalized Hamiltonian tools, *Q. J. R. Meteorol. Soc.* **134** (2008), 1597–1613.

[19] Germano M., Turbulence: the filtering approach, *J. Fluid Mech.* **238** (1992), 325–336.

[20] Oberlack M., *Invariant modeling in large-eddy simulation of turbulence*, *in: Annual research briefs*, Stanford University, 1997.

[21] Oberlack M., Similarity in non-rotating and rotating turbulent pipe flows, *J. Fluid Mech.* **379** (1999), 1–22.

[22] Oberlack M. and Rosteck A., New statistical symmetries of the multi-point equations and its importance for turbulent scaling laws, *Discrete Contin. Dyn. Syst. Ser. S* **3** (2010), 451–471.

[23] Olver P.J., *Application of Lie groups to differential equations*, Springer, New York, 2000.

[24] Olver P.J., *Equivalence, invariants and symmetry*, Cambridge University Press, Cambridge, 2009.

[25] Ovsiannikov L.V., *Group analysis of differential equations*, Acad. Press, New York, 1982.

[26] Popovych R.O. and Bihlo A., Symmetry preserving parameterization schemes, *J. Math. Phys.* **53** (2012), 073102 (36 pages), arXiv:1010.3010.

[27] Popovych R.O., Kunzinger M. and Eshraghi H., Admissible transformations and normalized classes of nonlinear Schrödinger equations, *Acta Appl. Math.* **109** (2010), 315–359.

[28] Popovych R.O., Kunzinger M. and Ivanova N.M., Conservation laws and potential symmetries of linear parabolic equations, *Acta Appl. Math.* **100** (2008), 113–185.

[29] Razafindralandy D., Hamdouni A. and Oberlack M., Analysis and development of subgrid turbulence models preserving the symmetry properties of the Navier–Stokes equations, *Eur. J. Mech. B/Fluids* **26** (2007), 531–550.

[30] Sagaut P., *Large eddy simulation for incompressible flows*, vol. 20 of *Scientific computation*, Springer, Berlin, 2005.

[31] Sommer M. and Névir P., A conservative scheme for the shallow-water system on a staggered geodesic grid based on a Nambu representation, *Q. J. R. Meteorol. Soc.* **135** (2009), 485–494.

[32] Sommer M. and Reich S., Phase space volume conservation under space and time discretization schemes for the shallow-water equations, *Mon. Wea. Rev.* **138** (2010), 4229–4236.

Part II

Structure-preserving integration

Chapter 4

Invariant discretization schemes for the shallow-water equations

Alexander Bihlo[†] and Roman O. Popovych[‡]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*
E-mail*: bihlo@crm.umontreal.ca*

[‡] *Faculty of Mathematics, University of Vienna, Nordbergstraße 15, A-1090 Vienna, Austria*
Institute of Mathematics of NAS of Ukraine, 3 Tereshchenkivska Str., 01601 Kyiv, Ukraine
E-mail*: rop@imath.kiev.ua*

Invariant discretization schemes are derived for the one- and two-dimensional shallow-water equations with periodic boundary conditions. While originally designed for constructing invariant finite difference schemes, we extend the usage of difference invariants to allow constructing of invariant finite volume methods as well. It is found that the classical invariant schemes converge to the Lagrangian formulation of the shallow-water equations. These schemes require to redistribute the grid points according to the physical fluid velocity, i.e. the mesh cannot remain fixed in the course of the numerical integration. Invariant Eulerian discretization schemes are proposed for the shallow-water equations in computational coordinates. Instead of using the fluid velocity as the grid velocity, an invariant moving mesh generator is invoked in order to determine the location of the grid points at the subsequent time level. The numerical conservation of energy, mass and momentum is evaluated for both the invariant and non-invariant schemes.

4.1 Introduction

Discretization schemes that preserve characteristic properties of systems of differential equations received an increasing attention over the past years and led to the development of the field of *geometric numerical integration*. The principal motivation for this approach is that controlling the local discretization error, as is done in most of the classical numerical methods, can fail to capture essential qualitative features of the underlying problem, which might be equally important in order to obtain reasonable integration results. Such features can include, but are not necessarily exhausted by, conservation laws, point symmetries, a Hamiltonian structure, conservation

of phase-space volume and asymptotic characteristics. Various geometric numerical integration schemes have been developed that capture these properties in the course of discretization, such as conservation laws and the Hamiltonian structure [4, 33, 43], Lie symmetries [2, 6, 9, 10, 12, 52] and phase-space volume [26, 47].

In the present paper, we aim to concentrate on the problem of deriving discretization schemes with symmetry properties by developing appropriate finite difference and finite volume schemes for the shallow-water equations. In particular, we are concerned with the problem of finding discretization schemes that are invariant under the maximal Lie invariance group admitted by the shallow-water equations with (double) periodic boundary conditions. Choosing the shallow-water equations for such an investigation can be motivated as they constitute a prominent, simple, yet fully-nonlinear model of fluid mechanics exhibiting various features of the original set of governing equations of hydrodynamics, such as the simultaneous occurrence of both fast (divergent) and slow (vortical) waves, the existence of conservation laws, symmetries and a Hamiltonian form. Moreover, the shallow-water equations always served as an important intermediate model to test new numerical schemes [2, 28, 32, 43–45].

Similarly as conservation laws, symmetries have important implications on the solutions of differential equations. When simulating the dynamics of a classical mechanical system in a constantly moving coordinate system, it should be a clear desire that the numerical model to be used for that problem is Galilean invariant as otherwise physical laws can be violated. It is also well known [8, 10, 11] that the shape of a solution of a system of differential equation near a blow-up point can tend to a group-invariant solution of this system. Often group-invariant solutions well describe so-called intermediate asymptotic behavior of the solutions after a sufficiently long period of evolution. For the simulation of invariant solutions, symmetry-preserving discretization schemes can give better numerical results than standard schemes that do not preserve the geometry of differential equations.

The design of invariant discretization schemes for evolution equations in general requires the explicit treatment of meshes that are not time-space orthogonal, i.e. time-adaptive grids. Such grids pose several challenges from the numerical point of view that are up to now not well investigated in the field of invariant numerical schemes. On the other hand, meshes that adapt according to the development of the numerical solution are an extensively investigated subject in the field of numerical mathematics, see e.g. [11, 49]. The question not explicitly answered so far is whether the problem of finding discretization schemes with symmetry properties can be embedded into the study of adaptive numerical schemes in the multidimensional case. In the present paper, we discuss a possible answer to that problem, exemplified with the shallow-water equations.

The outline of the paper is as follows: Properties of the shallow-water equations are discussed in Section 4.2. Section 5.2 is devoted to a review of common techniques that allows one to construct invariant finite difference schemes. In Section 4.4 we derive invariant discretization schemes for the one-dimensional shallow-water equations. This is done both by using the Lagrangian description of the shallow-water equations and by setting up an invariant grid generator for Eulerian schemes in computational coordinates. In Section 4.5 we discuss strategies for the design of invariant numerical models in higher dimensions and illustrate them with the two-dimensional shallow-water equations. Again, both Lagrangian schemes in the physical coordinates and Eulerian schemes in computational coordinates with an invariant grid generator are introduced. For the first scheme we use an invariant finite volume discretization, while the

second scheme is based on finite differences. A summary and concluding remarks can be found in the final Section 4.6.

4.2 Symmetries of the shallow-water equations

The nondimensionalized system of shallow-water equations in Cartesian coordinates is

$$\begin{aligned} &u_t + uu_x + vu_y + h_x = 0,\\ &v_t + uv_x + vv_y + h_y = 0,\\ &h_t + uh_x + vh_y + h(u_x + v_y) = 0, \end{aligned} \tag{4.1}$$

where $\mathbf{v} = (u, v)$ is the fluid velocity in the plane and h is the height of the fluid column over a fixed reference level within the fluid. The bottom topography is assumed to be flat here for simplicity. Treating non-flat topographies would lead to the inclusion of additional source terms in system (4.1). The shallow-water equations are derived from the Euler equations for an ideal fluid under the assumptions of validity of the hydro-static approximation, constance of the fluid density and that vertical motions are of much smaller scale than horizontal motions [41].

The shallow-water equations (4.1) can be represented in Hamiltonian form [35] upon using

$$\{\mathcal{F}, \mathcal{G}\} = \int \left(q\mathbf{k} \cdot \frac{\delta \mathcal{F}}{\delta \mathbf{v}} \times \frac{\delta \mathcal{G}}{\delta \mathbf{v}} - \frac{\delta \mathcal{F}}{\delta \mathbf{v}} \cdot \nabla \frac{\delta \mathcal{G}}{\delta h} + \frac{\delta \mathcal{G}}{\delta \mathbf{v}} \cdot \nabla \frac{\delta \mathcal{F}}{\delta h} \right) \mathrm{d}A$$

as Poisson bracket, where $\mathcal{F}$ and $\mathcal{G}$ are functionals of $\mathbf{v}$ and h, $q = \zeta/h = (v_x - u_y)/h$ is the potential vorticity, $\mathbf{k}$ denotes the vertical unit vector, $\mathrm{d}A = \mathrm{d}x\mathrm{d}y$ is the area element and the integration extends over the domain of the entire fluid. The Hamiltonian for the shallow-water equations is given by the total energy

$$\mathcal{H} = \frac{1}{2} \int \left(h\mathbf{v}^2 + h^2 \right) \mathrm{d}A.$$

Additional conserved quantities are associated with the above non-canonical Poisson bracket. For any function f of the potential vorticity q, the integral

$$\mathcal{C}_f = \int h f(q) \, \mathrm{d}A$$

is conserved on solutions of the shallow-water equations. This class of conserved quantities contains the mass $\mathcal{M} = \mathcal{C}_1$, the circulation $\mathcal{Z} = \mathcal{C}_q$ and the potential enstrophy $\mathcal{E} = \mathcal{C}_{q^2/2}$. Two more conserved quantities are the momenta in the x- and y-directions,

$$\mathcal{P}_x = \int hu \, \mathrm{d}A, \quad \mathcal{P}_y = \int hv \, \mathrm{d}A.$$

The maximal Lie invariance algebra $\mathfrak{g}_2$ of the two-dimensional shallow-water equations (4.1) is nine-dimensional, see e.g. [19, 40]. A basis of this algebra consists of the vector fields

$$\begin{aligned} &\partial_t, \quad \partial_x, \quad \partial_y, \quad t\partial_x + \partial_u, \quad t\partial_y + \partial_v,\\ &t\partial_t + x\partial_x + y\partial_y, \quad x\partial_x + y\partial_y + u\partial_u + v\partial_v + 2h\partial_h,\\ &-y\partial_x + x\partial_y - v\partial_u + u\partial_v, \quad t^2\partial_t + tx\partial_x + ty\partial_y + (x - tu)\partial_u + (y - tv)\partial_v - 2th\partial_h. \end{aligned}$$

These vector fields generate one-parametric Lie symmetry groups, which correspond to: (i) time-translations, (ii)–(iii) space translations, (iv)–(v) Galilean transformations, (vi)–(vii) scalings, (viii) rotations and (ix) inversions in t.

In what follows, we will also use the one-dimensional version of system (4.1), in which case we set $v = 0$ and drop the dependence of u and h on y. The resulting system reads

$$u_t + uu_x + h_x = 0, \quad h_t + uh_x + hu_x = 0, \tag{4.2}$$

and preserves the one-dimensional versions of total energy, mass and momentum,

$$\mathcal{H} = \frac{1}{2}\int \left(hu^2 + h^2\right) \mathrm{d}x, \qquad \mathcal{M} = \int h \,\mathrm{d}x, \qquad \mathcal{P} = \int hu \,\mathrm{d}x.$$

It is well known that the maximal Lie invariance algebra $\mathfrak{g}_1$ of system (4.2) is infinite dimensional and spanned by the vector fields

$$t\partial_t + x\partial_x, \quad x\partial_x + u\partial_u + 2h\partial_h, \quad t\partial_x + \partial_u,$$
$$(2x - 6tu)\partial_t + (6h - 3u^2)t\partial_x + (u^2 + 4h)\partial_u + 4hu\partial_h, \quad f(h,u)\partial_t + g(h,u)\partial_x,$$

where the functions f and g run through the set of solutions of the system

$$g_h - uf_h + f_u = 0, \quad g_u - uf_u + hf_h = 0. \tag{4.3}$$

The existence of the latter generator is owed to the possibility of linearization of system (4.2) to system (4.3) by means of the hodograph transformation in which u and h are assumed as the new independent variables and $f = t$ and $g = x$ are the new unknown functions. The linearization by the hodograph transformation permuting the pairs of dependent and independent variables is a general property of homogeneous first-order systems of partial differential equations in two independent variables and two unknown functions that are linear in derivatives with coefficients depending only on the unknown functions. See also [30, p. 154] for the symmetry interpretation of the linearization of the one-dimensional shallow-water equations. Note that system (4.3) is reduced to a single Tricomi equation. More precisely, excluding g by cross differentiation, we obtain the equation $f_{uu} = hf_{hh} + 2f_h$. The substitution $\phi = hf$ then leads to the Tricomi equation $\phi_{uu} = h\phi_{hh}$. Another way for the reduction is to rewrite the equation $f_{uu} = hf_{hh} + 2f_h$ in the form $h^3 f_{uu} = h^2(h^2 f_h)_h$ and to carry out the transformation $z = 1/h$ which yields the similar Tricomi equation $f_{uu} = z^3 f_{zz}$. Symmetry analysis of such equations was carried out in [7, 9].

4.3 Construction of invariant numerical discretization schemes

The problem of constructing discretization schemes that preserve symmetries of the corresponding differential equations was first systematically addressed by Dorodnitsyn and his collaborators [2, 6–8, 8, 21]. As there are an infinite number of possibilities to approximate a differential equation by means of finite differences, one might single out those among all possible difference schemes which inherit symmetries of the original differential equations. Dorodnitsyn's approach can be subsumed in the following way: First determine the maximal Lie invariance algebra of the model under consideration. For many classical hydrodynamical problems, this task is already completed [30, 31]. Hereafter, a discretization stencil has to be chosen. The generators of Lie symmetries are then prolonged to all points of the stencil. From these prolonged generators, the invariants of the extended group action on the stencil are determined. The final step then

is to assemble the obtained invariants together to a difference approximation of the original differential equation. By difference approximation it is meant that in the continuous limit, the invariant finite difference scheme reduces to the original differential equation. In much the same way, also the equations governing the positions of the grid points are composed in an invariant way. In the continuous limits, these grid equations reduce to some trivial identities.

Altogether, this method is a straightforward application of *inverse group classification*, using transformation groups acting on functions defined on a discrete set of points rather than on a continuous space. In the usual inverse group classification one starts with a particular Lie group G and aims at finding those systems of differential equations that admit G as a symmetry group. In practice those systems are found by computing differential invariants (i.e. invariants that involve derivatives of dependent variables) of G. Any function of differential invariants is a differential invariant of G and, subject to some regularity condition, any system of differential equations can be expressed in terms of differential invariants of its maximal Lie invariance group [38]. The Dorodnitsyn method works by selecting the maximal Lie symmetry group of a system of differential equations as the initial Lie group G. By extending the action of G to the points of the discretization stencil one is able to compute invariants of the extended action, i.e. difference invariants of G. As in the continuous case, any function of difference invariants is a difference invariant. Constructing a difference approximation of a system of differential equations using difference invariants therefore leads to a symmetry-preserving discretization scheme.

A common feature of difference schemes constructed by the above method is that grid points might not remain fixed in the course of the numerical integration. Precise criteria for a grid to be uniform, orthogonal and possessing flat time layers are formulated as conditions on coefficients of infinitesimal symmetry generators and are broken, e.g. for Galilean boosts and inversions [6]. This means that for such symmetries it is not possible to use isotropic or static grids. Hence, the problem of establishing good conditions governing the position of grid points both spatially and temporally becomes vital.

Up to now the reviewed technique has been applied to physically rather simple models, usually only involving time and one space dimension [2, 8, 8, 21, 52]. It is understandable that the multidimensional case is even more delicate, as there is an increasing number of possibilities for assembling the difference invariants to finite difference schemes. In addition, grids can evolve differently in distinct spatial dimensions, which might cause severe numerical problems if not treated appropriately, such as tangling meshes. In the present paper, we aim to discuss ways of overcoming the latter problem.

An alternate approach of constructing finite difference schemes with symmetry properties uses moving frames in the Fels and Olver formulation [9]. In contrast to the Dorodnitsyn approach, where finite difference schemes are constructed from the outset, in the moving frame method the concept of *invariantization* of existing schemes plays the key role. This technique can be summed up as follows [12, 22]: Determine the Lie symmetry group of the given system of differential equations. This part is standard and usually involves exponentiation of elements of the maximal Lie invariance algebra of the system. Subsequently, a moving frame associated with the Lie symmetry group is constructed. Roughly speaking, the moving frame is an equivariant function that returns the unique group element mapping a given point to an element of a chosen submanifold (the cross-section), which intersects each group orbit once and transversally. Since the conditions for a submanifold to be a cross-section to the group orbits are quite general, there is a freedom in choosing it and hence in constructing the associated moving frame. Once the

moving frame is obtained, it can be used to map an arbitrary function to an invariant function. This is a general property of any moving frame which allows determining usual invariants and differential invariants of a group action [18, 36]. In the same way, it is possible to take a given difference scheme (considered as a function of grid points) for a system of differential equations and apply the moving frame to it. By this procedure, the given scheme is transformed to a new scheme that will be invariant under the same Lie group as has been used to determine the moving frame.

The main benefit of this method is the possibility of using existing finite difference schemes as a starting point in the development of invariant schemes. Consequently, such invariant schemes could eventually be implemented into existing numerical models with limited effort. At the same time, the freedom in constructing a moving frame can make it somewhat difficult to predict the precise form of invariantized expressions and, therefore, to arrive at a scheme that is not only invariant but also has some desirable numerical properties, as e.g. discussed in [9]. Although difference invariants can be assembled to invariant discretization schemes in a variety of ways, the form of the particular difference invariants usually imposes enough hints in order to find reasonable finite difference approximations of a given system of differential equations. As with the Dorodnitsyn method described above, invariantization of existing numerical schemes may also lead to grids that evolve during numerical integration.

Another method for the construction of schemes with certain invariance properties was proposed in [7] for equations describing blow-up problems, see also [8, 10, 11]. The main idea in this approach is to use adaptive moving meshes at once because they are well suited for problems that develop shocks after a finite integration time. As a moving mesh complicates the discretization of differential equations in the physical space, the system to be discretized is first transformed into so-called *computational coordinates* that remain orthogonal and do not evolve during the numerical integration. The physical system is then discretized in the computational coordinates. It is advocated in this approach that for equations exhibiting blow-ups and for the description of the solution near the singularity, scale invariance plays an exceptional role. Therefore, scale invariance is required to be preserved in the course of discretization. The evolution of the mesh is formulated as an auxiliary system of differential equations, the so-called moving mesh partial differential equations. The auxiliary system is then selected in such a manner so as to preserve the scale invariance of the original physical model. A straightforward extension to the above approach is to require the mesh equations to not only possess the scale symmetries but also the other symmetries that the system of physical differential equations admits. The following property is basic for this extension: *The prolongation of any point symmetry of the initial system $\mathcal{L}$ of differential equations to the computational coordinates by means of the identical transformation is a point symmetry of the counterpart of $\mathcal{L}$ in terms of the computational coordinates.*

In the present paper, we will introduce yet one further approach for the construction of invariant discretization schemes which will be essential for multidimensional systems of differential equations. It rests on first expressing the system of differential equations under consideration in terms of computational coordinates and then extending the symmetry transformations of the original system to the system written in computational variables. Once it is understood how the system behaves under the extended symmetry transformations, one constructs a finite difference scheme that is transformed by the discretized version of the extended transformations in a similar way. In addition, the extra differential equations that control the location of the grid points are discretized in an invariant way, e.g. by using the finite difference invariants.

The reason why it is necessary to develop one more technique for the construction of invariant discretization schemes is twofold. Firstly, it is rather difficult to set up a proper invariant scheme for systems of differential equation using difference invariants as basic building blocks. For multidimensional moving meshes an additional problem is to find proper finite difference analogs of derivatives. Secondly, it can be (and, in general, will be) desirable to include additional qualitative properties of differential equations in the construction of invariant discretization schemes. Within the invariantization technique it might be tedious to ensure the numerical preservation of certain conservation laws, even if the initial system includes equations represented as conservation laws, which is precisely the case for the shallow-water equations written in momentum form. A similar remark holds for the Dorodnitsyn approach. An exception is given for equations derived from a variational principle for which, in view of the discrete version of the Noether identity, conservation laws and symmetries can be simultaneously preserved in the course of a proper invariant discretization of the associated Lagrangian [8]. There is still no algorithm using only difference invariants (or the invariantization map) that guarantees that the resulting invariant scheme will admits certain conservation laws. On the other hand, with the new approach to be introduced in the present paper, it is possible to construct, in a quite direct way, schemes that are both invariant and preserve some of the conservation laws possessed by the initial system.

4.4 Invariant numerical models for the one-dimensional shallow-water equations

In Section 4.2 we discussed the Lie symmetries of the shallow-water equations without any relation to boundary value problems. However, when setting up a numerical model for a specific set of problems, the explicit treatment of certain boundary conditions is usually inevitable. As a rule, the Lie symmetries possessed by a boundary value problem form only a subgroup (often even trivial one) of the maximal Lie symmetry group admitted by the involved system of differential equations [4]. Stated in another way, a specific boundary value problem usually admits only a small subset of the symmetries of the associated system of differential equations considered without boundary and initial conditions. This is in particular the case for various differential equations arising in hydrodynamics, which admit wide Lie invariance groups in the absence of boundary and initial conditions [1, 4–6, 30, 31]. This is also the case for the shallow-water equations as shown in the present section. Therefore, it is necessary to find a way to incorporate the boundary and initial conditions considered into the numerical model to be developed.

4.4.1 Selection of symmetries using boundary conditions

In order to design invariant numerical schemes for a system of differential equations, two principal strategies can be adopted.

In the first approach, which is applied in most of the previous works on invariant discretization [6], numerical schemes preserving the entire maximal Lie invariance groups of the corresponding systems are developed and then implemented for the specific physical configurations of interest. The drawback of this approach is that the practical implementation of a numerical scheme always requires an explicit treatment of a boundary value problem. As was said above, for the boundary conditions arising in hydrodynamics most often (e.g. periodic, reflective or absorbing), the maximal Lie invariance groups of systems without boundary

conditions are usually much wider than those of particular boundary value problems. Using the first approach may therefore lead to the overly restrictive requirement that all the symmetries of the considered system of differential equation are equally important in the course of invariant discretization. For a particular boundary value problem, however, this may not be the case, as some of the symmetries of the differential equation might not be admitted at all.

This is why we adopt the second approach here, which only requires the preservation of symmetries that are compatible with the class of specific boundary value problems under consideration. The apparent drawback of this approach is that, if one aims to test different kinds of boundary conditions, it can be necessary to design a new scheme for each configuration, as different symmetry groups may arise under varying specific settings. On the other hand, for a model to be used for a particular purpose (e.g. a weather or climate prediction model) the boundary conditions are generally fixed at the stage of model development and therefore do not change subsequently. Another advantage of the second approach is of more physical nature. The restriction imposed on symmetries in that they map a given boundary value problem to itself is too restrictive even from the physical perspective. When transforming a given reference frame to another reference frame, the boundary value problems of the reference frames involved are also mapped to each other. Therefore, it is not natural to require appropriate symmetries of a system of differential equations to preserve a particular boundary value problem but rather to only impose that these symmetries map boundary value problems from a class of such problems (e.g. periodic domains of *any* size with varying initial time and initial conditions) to each other. Such transformations are known as *equivalence transformations* and when deriving symmetry-preserving discretization schemes, we require a subgroup of the maximal Lie invariance group of the original system of differential equation to be compatible with the structure of a predefined class of boundary value problems, i.e. elements of the subgroup should act as equivalence transformations on the boundary conditions rather than as symmetry transformations.

The relaxed condition of requiring the finite difference schemes to be invariant only under the transformations admitted by a class of boundary value problems thus provides a natural selection criterion for subgroups of a maximal Lie invariance group to be preserved numerically. In view of the particular nature of the infinite-dimensional maximal Lie invariance algebra $\mathfrak{g}_1$ of the one-dimensional shallow-water equations (or, more generally, systems of differential equations arising in hydrodynamics), it might be a cumbersome or even useless task to attempt preserving all these symmetries in the respective discrete models.

Among the most natural boundary conditions for the one-dimensional shallow-water equations in the setting of geophysical fluid dynamics are periodic ones, which we aim to study here. These boundary conditions are advantageous as they are not as restrictive as, e.g., Dirichlet boundary conditions from the pure symmetry point of view and generally lead to the selection of symmetries that have a clear physical interpretation. The subalgebra $\mathfrak{s}_1$ of $\mathfrak{g}_1$ which is compatible with periodic boundary conditions is spanned by the vector fields

$$\partial_t, \quad \partial_x, \quad t\partial_x + \partial_u, \quad t\partial_t + x\partial_x, \quad x\partial_x + u\partial_u + 2h\partial_h. \tag{4.4}$$

Even if we would neglect initial conditions, only elements from the narrower subalgebra $\langle \partial_t, \partial_x, t\partial_x + \partial_u, t\partial_t - u\partial_u - 2h\partial_h \rangle$ generate one-parametric symmetry groups of such a boundary value problem as scalings with respect to x are not admitted once a domain (periodicity) length is fixed. However, the inclusion of scaling symmetries in the subsequent consideration is justified as they

are equivalence transformations of the chosen class of boundary value problems. In other words, upon preserving the subalgebra $\mathfrak{s}_1$ we are still able to test different domain lengths. In this sense, the preservation of scaling transformations plays an important role for the class of boundary value problems for the one-dimensional shallow-water equations with periodic boundary conditions and *any* domain size even if scalings are no proper symmetry transformations for a specific numerical integration.

4.4.2 Classical invariant schemes and beyond

We begin our study of invariant numerical schemes for the shallow-water equations using the classical construction proposed by Dorodnitsyn. Within this framework, we have to prolong the selected subalgebra $\mathfrak{s}_1$ to the discretization stencils which we aim to use. These stencils are depicted in Fig. 4.1. As the symmetry group associated with $\mathfrak{s}_1$ does not violate the criterion for using flat time layers, see [6] for more details, all points in the spatial domain are defined at the same time. However, preserving Galilean invariance in a numerical scheme it is impossible to use a fixed grid, i.e. $\hat{x}_i \neq x_i$. Here and in what follows variables with hat and without hat denote values on the grid at the time levels $t+\tau$ and t, respectively, and τ is the time step. The possibility of evolving grids in general also leads to nonhomogeneous spacings in the course of the integration, i.e. $\hat{x}_{i+1} - \hat{x}_i \neq \hat{x}_i - \hat{x}_{i-1}$ even if the initial grid $\{x_i\}$ is equally spaced.

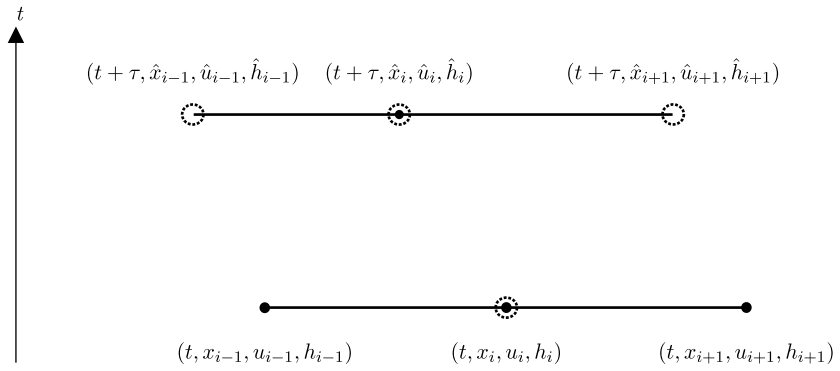

Figure 4.1: Stencils for invariant schemes of the one-dimensional shallow-water equations. An explicit (Euler forward) and an implicit (Euler backward) scheme is defined using the points indicated by filled and dashed circles, respectively.

Prolonging the vector fields (4.4) to the points indicated in Fig. 4.1 gives

$$
\begin{aligned}
&\partial_t, \quad \partial_{x_i} + \partial_{x_{i+1}} + \partial_{x_{i-1}} + \partial_{\hat{x}_i} + \partial_{\hat{x}_{i+1}} + \partial_{\hat{x}_{i-1}},\\
&t(\partial_{x_i} + \partial_{x_{i+1}} + \partial_{x_{i-1}}) + (t+\tau)(\partial_{\hat{x}_i} + \partial_{\hat{x}_{i+1}} + \partial_{\hat{x}_{i-1}}) +\\
&\partial_{u_i} + \partial_{u_{i+1}} + \partial_{u_{i-1}} + \partial_{\hat{u}_i} + \partial_{\hat{u}_{i+1}} + \partial_{\hat{u}_{i-1}},\\
&t\partial_t + \tau\partial_\tau + x_i\partial_{x_i} + x_{i+1}\partial_{x_{i+1}} + x_{i-1}\partial_{x_{i-1}} + \hat{x}_i\partial_{\hat{x}_i} + \hat{x}_{i+1}\partial_{\hat{x}_{i+1}} + \hat{x}_{i-1}\partial_{\hat{x}_{i-1}},\\
&x_i\partial_{x_i} + x_{i+1}\partial_{x_{i+1}} + x_{i-1}\partial_{x_{i-1}} + \hat{x}_i\partial_{\hat{x}_i} + \hat{x}_{i+1}\partial_{\hat{x}_{i+1}} + \hat{x}_{i-1}\partial_{\hat{x}_{i-1}} +\\
&u_i\partial_{u_i} + u_{i+1}\partial_{u_{i+1}} + u_{i-1}\partial_{u_{i-1}} + \hat{u}_i\partial_{\hat{u}_i} + \hat{u}_{i+1}\partial_{\hat{u}_{i+1}} + \hat{u}_{i-1}\partial_{\hat{u}_{i-1}} +\\
&2h_i\partial_{h_i} + 2h_{i+1}\partial_{h_{i+1}} + 2h_{i-1}\partial_{h_{i-1}} + 2\hat{h}_i\partial_{\hat{h}_i} + 2\hat{h}_{i+1}\partial_{\hat{h}_{i+1}} + 2\hat{h}_{i-1}\partial_{\hat{h}_{i-1}}.
\end{aligned}
\tag{4.5}
$$

To construct an explicit (Euler forward) numerical scheme, we have to restrict ourselves in (4.5) to the values which are defined at the filled circles depicted in Fig. 4.1. A convenient complete set of functionally independent difference invariants then is

$$
\begin{aligned}
&I_0=\frac{x_{i+1}-x_i}{x_i-x_{i-1}},\quad I_1=\frac{\dot x_i-u_i}{x_{i+1}-x_{i-1}}\tau,\quad I_2=\frac{\hat u_i-u_i}{x_{i+1}-x_{i-1}}\tau,\\
&I_3=\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}\tau,\quad I_4=\frac{u_{i+1}-u_i}{x_{i+1}-x_i}\tau,\quad I_5=\frac{h_{i-1}}{(x_{i+1}-x_{i-1})^2}\tau^2,\\
&I_6=\frac{h_i}{(x_{i+1}-x_{i-1})^2}\tau^2,\quad I_7=\frac{h_{i+1}}{(x_{i+1}-x_{i-1})^2}\tau^2,\quad I_8=\frac{\hat h_i}{(x_{i+1}-x_{i-1})^2}\tau^2,
\end{aligned}
\tag{4.6}
$$

where $\dot x_i=(\hat x_i-x_i)/\tau$ by definition is the mesh velocity. These invariants are found from integrating the system of first-order quasilinear partial differential equations $\mathbf{v}_j(I)=0$, where $\mathbf{v}_j$, $j=1,\dots,5$, are the prolonged vector fields presented in (4.5). Such a system for invariants admits precisely $m_{\rm i}=m_{\rm s}-r$ functionally independent solutions, where $m_{\rm s}$ is the number of stencil variables and r is the rank of involved vector fields. For the explicit scheme, we have $m_{\rm s}=14$, $r=5$ and hence $m_{\rm i}=9$.

Using the difference invariants of the set (4.6) we can approximate (4.2) via

$$I_1=0,\quad I_2+I_7-I_5=0,\quad I_8-I_6+I_6I_3=0,$$

or, explicitly,

$$
\dot x_i=u_i,\quad \frac{\hat u_i-u_i}{\tau}+\frac{h_{i+1}-h_{i-1}}{x_{i+1}-x_{i-1}}=0,\quad \frac{\hat h_i-h_i}{\tau}+h_i\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}=0.
\tag{4.7}
$$

In the continuous limit the above scheme leads to the following system of differential equations

$$
\frac{\mathrm{d}x}{\mathrm{d}t}=u,\quad \frac{\mathrm{d}u}{\mathrm{d}t}+\frac{\partial h}{\partial x}=0,\quad \frac{\mathrm{d}h}{\mathrm{d}t}+h\frac{\partial u}{\partial x}=0,
\tag{4.8}
$$

which is (4.2) in Lagrangian variables.

As the Euler forward scheme is conditionally stable, it is beneficial to construct an implicit invariant numerical scheme. A simple implicit scheme is the Euler backward scheme and it can be constructed in a similar way as the invariant Euler forward scheme. However, we prefer to at once construct a trapezoidal scheme, which has in general a greater accuracy. To accomplish this we additionally need the difference invariants

$$
I_9=\frac{\dot x_i-\hat u_i}{x_{i+1}-x_{i-1}}\tau,\quad I_{10}=\frac{\hat u_{i+1}-\hat u_{i-1}}{\hat x_{i+1}-\hat x_{i-1}}\tau,\quad I_{11}=\frac{\hat h_{i+1}-\hat h_{i-1}}{(x_{i+1}-x_{i-1})(\hat x_{i+1}-\hat x_{i-1})}\tau^2.
\tag{4.9}
$$

Note that $\{I_0,\dots,I_{11}\}$ is not a complete set of functionally independent invariants for the transformation group generated by the vector fields (4.5) on the trapezoidal stencil as the total number of variables on this stencil equals 20 and hence a functional basis of related invariants consists of 15 invariants. By combining the invariants I_1–I_3 and I_5–I_{11} we construct

$$I_1+I_9=0,\quad I_2+\frac12(I_7-I_5+I_{11})=0,\quad I_8-I_6+\frac12(I_6I_3+I_8I_{10})=0,$$

which boils down to the form

$$
\begin{aligned}
&\dot x_i=\frac12(u_i+\hat u_i),\\
&\frac{\hat u_i-u_i}{\tau}+\frac12\left(\frac{h_{i+1}-h_{i-1}}{x_{i+1}-x_{i-1}}+\frac{\hat h_{i+1}-\hat h_{i-1}}{\hat x_{i+1}-\hat x_{i-1}}\right)=0,\\
&\frac{\hat h_i-h_i}{\tau}+\frac12\left(h_i\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}+\hat h_i\frac{\hat u_{i+1}-\hat u_{i-1}}{\hat x_{i+1}-\hat x_{i-1}}\right)=0.
\end{aligned}
\tag{4.10}
$$

This scheme also converges to (4.8).

The problem with the above schemes in particular and with the difference invariants approach to the construction of invariant schemes in general is that it is hard to control properties other than symmetries which the resulting discretizations admit. It can be checked by direct computation that the above schemes violate even the mass conservation law (conservation of momentum and energy is violated as well). This violation of fundamental conservation laws is a direct consequence of the construction method of the invariant finite difference schemes, which only takes into account local information on u_i and h_i (i.e. the difference invariants) but provides no guideline ensuring the preservation of global feature by the numerical solution.

In the present case, this problem can be partially circumvented by discretizing the shallow-water equations not in the form (4.8) but rather in the momentum form

$$\mathrm{Eq}^u = (uh)_t + \left(hu^2 + \frac{1}{2}h^2\right)_x = 0, \quad \mathrm{Eq}^h = h_t + (uh)_x = 0. \tag{4.11}$$

An invariant finite difference approximation of this system using an Euler time step is

$$\dot{x}_i = u_i, \quad \hat{u}_i\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - u_ih_i + \frac{\tau}{2}\frac{h_{i+1}^2 - h_{i-1}^2}{x_{i+1} - x_{i-1}} = 0, \quad \hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - h_i = 0,$$

while the invariant trapezoidal scheme is

$$\begin{aligned} &\dot{x}_i = \frac{1}{2}(u_i + \hat{u}_i), \quad \hat{u}_i\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - u_ih_i + \frac{\tau}{4}\left(\frac{h_{i+1}^2 - h_{i-1}^2}{x_{i+1} - x_{i-1}} + \frac{\hat{h}_{i+1}^2 - \hat{h}_{i-1}^2}{\hat{x}_{i+1} - \hat{x}_{i-1}}\right) = 0,\\ &\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - h_i = 0. \end{aligned} \tag{4.12}$$

The schemes constructed this way numerically preserve the mass conservation law. The explicit scheme in addition preserves the momentum, while the implicit scheme (4.12) does not preserve momentum exactly, as $\hat{x}_{i+1} - \hat{x}_{i-1} \neq x_{i+1} - x_{i-1}$ in general. This condition, however, would be required to yield exact conservation of momentum but as the change in the spacings $x_{i+1} - x_{i-1}$ is not abrupt, the violation of momentum conservation of the scheme (4.12) is rather small, see the result of a numerical integration using (4.12) below. None of these schemes respects conservation of energy.

In the continuous limits both the explicit and implicit discretizations give

$$\frac{\mathrm{d}x}{\mathrm{d}t} = u, \quad \frac{\mathrm{d}(uh)}{\mathrm{d}t} + \frac{1}{2}\frac{\partial h^2}{\partial x} = 0, \quad \frac{\mathrm{d}h}{\mathrm{d}t} + h\frac{\partial u}{\partial x} = 0,$$

which is (4.11) expressed in Lagrangian variables. Similar as the schemes (4.7) and (4.10) the above two invariant schemes for the momentum form of the shallow-water equations could be expressed in terms of difference invariants. As these expressions are considerably more involved as the analog expressions for (4.7) and (4.10), we do not present these difference invariants forms here. The reason for the difference invariants expressions being more complicated in the present case can be traced back to the result of the Galilean transformation $\tilde{t} = t$, $\tilde{x} = x + \varepsilon_1 t$, $\tilde{u} = u + \varepsilon_1$, when applied to (4.11), which yields

$$\widetilde{\mathrm{Eq}}^h = \mathrm{Eq}^h, \quad \widetilde{\mathrm{Eq}}^u = \mathrm{Eq}^u + \varepsilon_1\mathrm{Eq}^h. \tag{4.13}$$

The transformation of the momentum equation thus yields a linear combination of the momentum equation with the continuity equation. This implies that the momentum equation itself

cannot be expressed in terms of differential invariants but only in combination with the continuity equation. It is thus not natural to approximate the momentum equation using difference invariants. At the same time, checking that the proposed conservative schemes are indeed invariant can be shown directly by acting with the prolonged vector fields (4.5) on them and verifying that the results of these operations yield zero on the solutions of the numerical scheme.

The result of a numerical integration taking harmonic initial conditions both for u and h using the scheme (4.12) is depicted in Fig. 4.2. As the evolution of the mesh points is directly coupled to the (initially harmonic) physical velocity, the single mesh points quasi-oscillate around their initial positions (Fig. 4.2a). No special ability of the mesh to follow the developing shock (Fig. 4.2b, showing the numerical solution of h at time $t = 3$) is visible, which is one of the major disadvantages of the scheme (4.12). The scheme conserves mass up to machine precision (10^{-16}) but it dissipates energy, with the relative change $(\mathcal{H}(t) - \mathcal{H}(0))/\mathcal{H}(0)$ of energy being of the order 10^{-5} at the end of the integration. The relative change in momentum in this integration is of order 10^{-14} without a positive or negative trend. The values of $\mathcal{M}$, $\mathcal{H}$ and $\mathcal{P}$ at time t are evaluated using the formulas

$$\mathcal{M} = \frac{1}{2}\sum_i h_i(x_{i+1} - x_{i-1}), \quad \mathcal{H} = \frac{1}{4}\sum_i (h_i u_i^2 + h_i^2)(x_{i+1} - x_{i-1}),$$
$$\mathcal{P} = \frac{1}{2}\sum_i h_i u_i (x_{i+1} - x_{i-1}),$$

respectively.

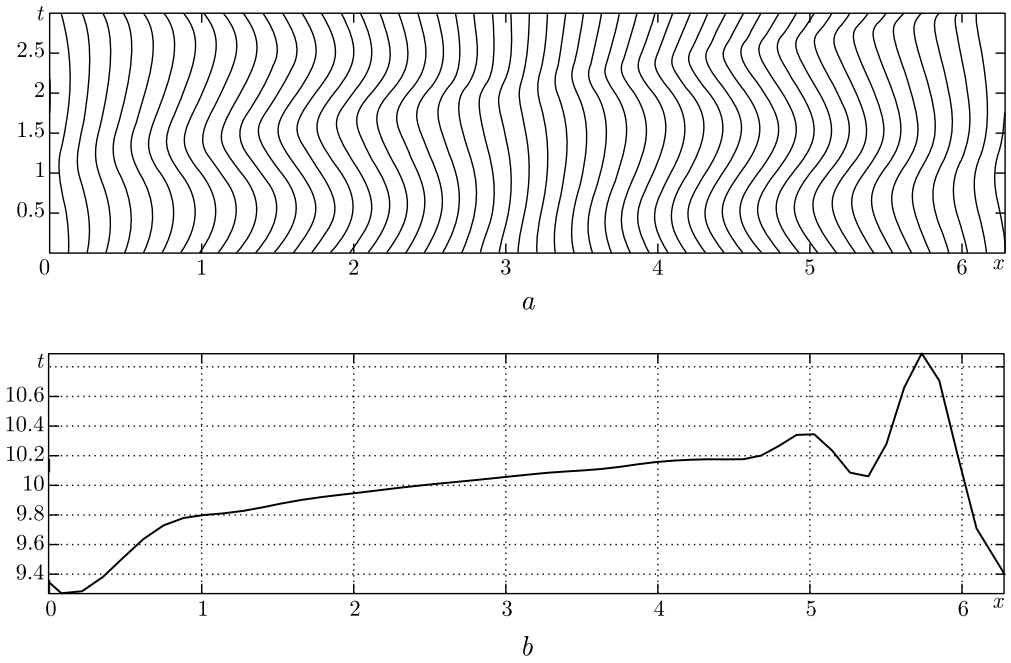

Figure 4.2: Numerical integration of the one-dimensional shallow-water equations (4.2) using the scheme (4.10) with $\tau = 0.001$ and $N = 51$ grid points on the domain $[0, 2\pi]$ over the time interval $[0, 3]$. The initial conditions are $u = A\sin x$, and $h = h_0 + A\sin(x + \varphi_0)$, with $A = 0.4$, $\varphi_0 = \pi/6$ and $h_0 = 10$. (a) Evolution of the discretization grid. (b) Numerical solution for h at $t = 3$.

4.4.3 Invariant discretization on equidistributing meshes

So far, we were mainly concerned with assembling difference invariants in a proper way, so as to guarantee the invariance of the resulting finite difference schemes. That is, the invariance condition was the relevant starting point in the design of the above schemes. The main problem with this approach is the lack of an explicit error control for the proposed numerical models. When setting up a numerical scheme, it is of primary interest to ensure not only the discrete preservation of qualitative features of differential equations. Also classical issues from numerical analysis have to be addressed. In the light of adaptive moving meshes, these issues mainly concern the prevention of abruptly changing grids, mesh racing and mesh tangling, which can significantly degrade the numerical solution and ultimately lead to convergence failure.

When dealing with finite difference schemes on adaptive moving meshes, one usually regards the mesh movement as a time-dependent coordinate transformation from a fixed logical (computational) domain to the physical domain of the system of differential equations. The computational coordinates are defined to index the positions of the grid points in the mesh. As in any regular grid each grid point keeps its position relative to its neighbors in the mesh even in the presence of adaption, it is convenient to take the (spacial) computational coordinates as time-independent, Cartesian and orthogonal, with uniform spacing on the unit interval (up to scaling). In the one-dimensional case considered here, the step of the spacial computational coordinate ξ equals $1/(N-1)$, where N is the number of grid points at a fixed time level. In order to use computational coordinates, it is necessary to transform the system of differential equations from the physical space to the index space, see e.g. [8, 11].

The relation of the usage of computational coordinates to invariant numerical schemes will be illustrated again with the one-dimensional system of shallow-water equations. The central idea is that any finite difference discretization of the shallow-water equations on a moving mesh in computational coordinates is invariant under the Lie group generated by the vector fields (4.5). Indeed, under the transformation $t=\theta$, $x=x(\theta,\xi)$ to the computational coordinates (θ,ξ), the one-dimensional shallow-water equations (4.2) takes the form

$$\tilde{u}_\theta+(\tilde{u}-x_\theta)\frac{\tilde{u}_\xi}{x_\xi}+\frac{1}{x_\xi}\tilde{h}_\xi=0,\quad \tilde{h}_\theta+(\tilde{u}-x_\theta)\frac{\tilde{h}_\xi}{x_\xi}+\frac{1}{x_\xi}\tilde{h}\tilde{u}_\xi=0, \tag{4.14}$$

where $\tilde{u}=u(\theta,x(\theta,\xi))$, $\tilde{h}=h(\theta,x(\theta,\xi))$. It is obvious that any usual finite difference discretization of (4.14) possesses the symmetry group requested. For example, discretizing using forward differences in time and central differences in space, from the above system we obtain

$$\begin{aligned}
&\frac{\hat{u}_i-u_i}{\tau}+(u_i-\dot{x}_i)\,\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}+\frac{h_{i+1}-h_{i-1}}{x_{i+1}-x_{i-1}}=0,\\
&\frac{\hat{h}_i-h_i}{\tau}+(u_i-\dot{x}_i)\,\frac{h_{i+1}-h_{i-1}}{x_{i+1}-x_{i-1}}+h_i\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}=0,
\end{aligned}$$

where $u_i=\tilde{u}(\theta,\xi_i)=u(t,x_i(t))$, $h_i=\tilde{h}(\theta,\xi_i)=h(t,x_i(t))$ and $\hat{u}_i$ and $\hat{h}_i$ denote the same values at $\theta+\tau$. This discretization coincides with the second and the third equation of the system (4.7) if we assume the grid evolution to be Lagrangian of the form $\dot{x}_i=u_i$.

In much the same way, an invariant implicit discretization (trapezoidal rule) can be obtained from (4.14), which reads

$$\begin{aligned}
&\frac{\hat{u}_i - u_i}{\tau} + \frac{1}{2}\left(\frac{u_i + \hat{u}_i}{2} - \dot{x}_i\right)\left(\frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} + \frac{\hat{u}_{i+1} - \hat{u}_{i-1}}{\hat{x}_{i+1} - \hat{x}_{i-1}}\right) + \\
&\frac{1}{2}\left(\frac{h_{i+1} - h_{i-1}}{x_{i+1} - x_{i-1}} + \frac{\hat{h}_{i+1} - \hat{h}_{i-1}}{\hat{x}_{i+1} - \hat{x}_{i-1}}\right) = 0, \\
&\frac{\hat{h}_i - h_i}{\tau} + \frac{1}{2}\left(\frac{u_i + \hat{u}_i}{2} - \dot{x}_i\right)\left(\frac{h_{i+1} - h_{i-1}}{x_{i+1} - x_{i-1}} + \frac{\hat{h}_{i+1} - \hat{h}_{i-1}}{\hat{x}_{i+1} - \hat{x}_{i-1}}\right) + \\
&\frac{1}{2}\left(h_i\frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} + \hat{h}_i\frac{\hat{u}_{i+1} - \hat{u}_{i-1}}{\hat{x}_{i+1} - \hat{x}_{i-1}}\right) = 0.
\end{aligned}$$

Again, in the Lagrangian case $\dot{x}_i = (u_i + \hat{u}_i)/2$, this scheme coincides with the scheme (4.10).

None of these two schemes preserves mass and momentum, as they approximate the representation (4.14) of the shallow-water equations, where the equations are not in conserved form. It is possible to discretize the conserved form (4.11) using computational variables as well, which boils down to

$$\hat{u}_i\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - u_ih_i + \tau A(uh) + \frac{\tau}{2}D(h^2) = 0, \quad \hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - h_i - \tau A(h) = 0,$$

for the explicit Euler scheme (preserving mass and momentum) and to

$$\begin{aligned}
&\hat{u}_i\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - u_ih_i + \frac{\tau}{2}(A(uh) + \hat{A}(uh)) + \frac{\tau}{4}(D(h^2) + \hat{D}(h^2)) = 0, \\
&\hat{h}_i\frac{\hat{x}_{i+1} - \hat{x}_{i-1}}{x_{i+1} - x_{i-1}} - h_i - \frac{\tau}{2}(A(h) + \hat{A}(h)) = 0,
\end{aligned} \tag{4.15}$$

for the implicit trapezoidal discretization. In both schemes we denote

$$A(z) = \frac{(u_{i+1} - \dot{x}_{i+1})z_{i+1} - (u_{i-1} - \dot{x}_{i-1})z_{i-1}}{x_{i+1} - x_{i-1}}, \quad D(z) = \frac{z_{i+1} - z_{i-1}}{x_{i+1} - x_{i-1}},$$

and $\hat{A}(z)$ and $\hat{D}(z)$ have the same forms as $A(z)$ and $D(z)$ with all the variables replaced by the associated variables on the next time step $\theta + \tau$, only keeping the grid velocity the same. In the continuous limit, these schemes converge to

$$(x_\xi F^t)_\theta + (F^x - F^t x_\theta)_\xi = 0,$$

which is indeed (4.11) in computational variables using $F^t = (h, hu)$ and $F^x = (hu, hu^2 + \frac{1}{2}h^2)$.

The above observation can be easily extended to other invariant schemes for evolution equations admitting Galilean transformations as symmetries. Its main benefit is that it allows us to establish a connection to the theory of discretization on adaptive moving meshes. This may aid in tackling the problem of finding invariant finite difference schemes which also have good numerical properties.

In order to complete the invariant schemes in computational coordinates, it is necessary to determine the mesh velocity $\dot{x}_i$ in an invariant way. This can be done using *equidistributing meshes*. Classically, a mesh is called equidistributed if the relation

$$\int_a^{x(\xi)} \rho(x)\mathrm{d}x = \xi\int_a^b \rho(x)\mathrm{d}x$$

holds for the continuous mapping $x = x(\xi)\colon [0,1] \to [a,b]$, see e.g. [11]. The function $\rho = \rho(x)$ is called the monitor function. It determines the regions of concentration of the grid. Differentiating this equation twice with respect to ξ, one obtains

$$(\rho x_\xi)_\xi = 0, \tag{4.16}$$

with the boundary conditions $x(0) = a$ and $x(1) = b$, which is satisfied for an equidistributed mesh. As we consider periodic boundary conditions, we should modify the classical framework of equidistributing meshes and replace the boundary conditions for $x(\xi)$ by setting $x(1)-x(0) = 2\pi$ and $x_\xi(0) = x_\xi(1)$. The periodic conditions for $x(\xi)$ are agreed with the invariance requested.

The above schemes in computational coordinates will therefore be completely invariant if we obtain the grid on the next level (and therefore the grid velocity $\dot{x}_i = (\hat{x}_i - x_i)/\tau$) from an invariant discretization of the equidistribution principle (7.37). The discretization

$$(\rho_{i+1} + \rho_i)(\hat{x}_{i+i} - \hat{x}_i) - (\rho_i + \rho_{i-1})(\hat{x}_i - \hat{x}_{i-1}) = 0, \tag{4.17}$$

is invariant provided that we choose an invariant monitor function ρ. An ansatz for ρ motivated from the theory of adaptive grids is, e.g., the arc-length(-like) monitor function

$$\rho = \sqrt{1 + \alpha u_x^2}$$

with α being the (positive) adaption constant. This monitor function is invariant with respect to vector fields (4.4) excluding only the scale operator $t\partial_t + x\partial_x$ but the corresponding scalings are equivalence transformations for the set of such monitor functions, where the parameter α varies. The above ansatz for ρ can be discretized in an invariant way via

$$\rho_i = \sqrt{1 + \alpha \left(\frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} \right)^2}. \tag{4.18}$$

The resulting form of Eq. (4.17) can then be solved either using an iterative method, such as Jacobi or Gauß–Seidel iteration, or by relaxation, e.g. using the moving mesh PDE approach [8, 11].

Remark 4.1. For the equation (7.37) to possess a Lie symmetry algebra $\mathfrak{g}$ which is contained in the linear span $\mathfrak{s}_1$ of vector fields (4.4) trivially extended to ξ, it suffices for the monitor function ρ to be an invariant of $\mathfrak{g}$. On solutions of the shallow-water equations (4.2) we can assume without loss of generality that the function ρ does not depend on derivatives of u and h involving differentiation with respect to t. Then the general form of ρ which is an invariant of the pure Galilean algebra $\langle \partial_t, \partial_x, t\partial_x + \partial_u \rangle$ is given by an arbitrary smooth function of derivatives of u and h with respect to x including h itself but not u. In order to attain invariance with respect to scale transformations, the function ρ should depend only on specific products of powers of the above derivatives. At the same time, the incorporation of geometric properties of solutions (e.g., the length of a graph between neighbouring grid points) to the monitor function is more important than scale invariance. Therefore, scale transformations can be allowed to act in a relaxed way, as equivalence transformations on a selected narrowed set of monitor functions. An obvious form that satisfies this requirement is the arc-length monitor function $\rho = \sqrt{1 + \alpha u_x^2}$. Alternatively one could use, e.g., the similar functions $\rho = \sqrt{1 + \alpha h_x^2}$ and $\rho = \sqrt{1 + \alpha u_x^2 + \beta h_x^2}$ or the curvature-related monitor functions $\rho = \sqrt{1 + \alpha u_{xx}^2}$, $\rho = \sqrt{1 + \alpha h_{xx}^2}$ and $\rho = \sqrt{1 + \alpha u_{xx}^2 + \beta h_{xx}^2}$, where α and β are positive constant.

Remark 4.2. The method for the construction of an invariant discretization of a differential equation in combination with a numerical grid generator was discussed, e.g., in [9, 10, 12]. In contrast to the method employed above, in [10] the space of stencil variables was also prolonged to the monitor function, which is not necessarily chosen in an invariant way. In order to arrive at a completely invariant model, we however regard it important that the equidistribution principle is discretized in an invariant fashion too, see also the discussion in Section 4.6. Moreover, as the monitor function involves independent variables, unknown functions and their derivatives, it is possible to express its discretization using the same basis difference invariants of stencil variables that is needed for the physical differential equation discretization. In other words, no explicit prolongation to the monitor function is necessary within the framework of our approach.

In Fig. 4.3 we show the integration of the one-dimensional shallow-water equations using the scheme (4.15), (4.17) with arc-length monitor function discretization (4.18) utilizing the same initial conditions as those chosen for the integration shown in Fig. 4.2. It is clearly visible that the mesh points almost remain fixed as long as the shock is not developed. Once the shock is traveling through the domain, the mesh points are able to sufficiently adapt to yield increased resolution in the region near the shock (as additionally shown in Fig. 4.3*c*). Again the scheme approximately conserves mass and momentum but dissipates energy. The relative errors in the momentum and energy conservation are approximately the same as in the case of the Lagrangian schemes in the previous subsection.

It is worthwhile pointing out that the time step of the integration shown in Fig. 4.3 is relatively small. The reason for this is that using the scheme (4.15) and (4.17)–(4.18) we decouple the solution of the physical differential equation and the equation controlling the location of grid points. If time steps are not small, a severe time lag in the mesh movement would occur and the resulting mesh would not satisfy the equidistribution principle close enough to give a satisfactory adaptivity. The above problem was extensively addressed in [11]. It can be overcome via the iterative solution of the physical and mesh equations a number of times, which leads to a reduction of the time lag in the mesh movement. Such a strategy could be readily adopted with the scheme (4.15) and (4.17)–(4.18) because a repeated iterative integration does not break the invariance of this scheme.

In order to facilitate the comparison of distinct types of invariant numerical schemes, we also keep the time step small in the integration of the Lagrangian scheme for the one-dimensional shallow-water equations which is presented in Fig. 4.2.

4.5 Invariant numerical models for the two-dimensional shallow-water equations

4.5.1 Selection of symmetries using boundary conditions

The domains most often considered in geophysical fluid dynamics for the numerical integration of the two-dimensional shallow-water equations on a plane are either a channel with periodic boundary conditions in the East–West direction and rigid boundaries in the North–South direction or a domain with double periodic boundary conditions. As the second configuration is more challenging from the point of view of invariant numerical schemes, we will employ it subsequently.

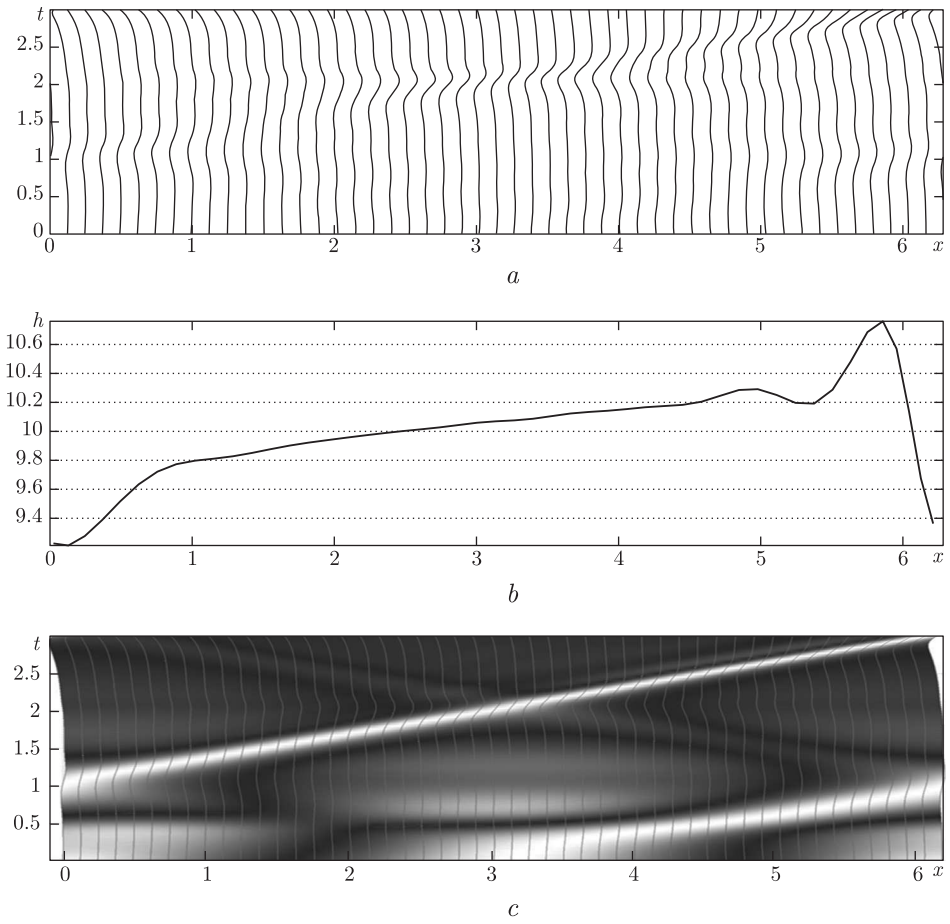

Figure 4.3: Numerical integration of the one-dimensional shallow-water equations (4.2) using the scheme (4.15) with $\tau = 0.001$ and $N = 51$ grid points on the domain $[0, 2\pi]$ over the time interval $[0, 3]$. The initial conditions are $u = A \sin x$, and $h = h_0 + A \sin(x + \varphi_0)$, with $A = 0.4$, $\varphi_0 = \pi/6$ and $h_0 = 10$. The trapezoidal rule is used for time integration and the arc-length monitor function is chosen for grid adaption setting $\alpha = 0.8$. (a) Evolution of the discretization grid. (b) Numerical solution for h at $t = 3$. (c) Magnitude of the derivative u_x of the solution for the scheme (4.15). Light colors refer to high values of $|u_x|$.

Lie symmetry operators of the two-dimensional shallow-water equations (4.1) with periodic boundary conditions in both the East–West and North–South directions form the five-dimensional subalgebra $\mathfrak{s}_2$ of the maximal Lie invariance algebra $\mathfrak{g}_2$ of the equations (4.1) without additional constraints. A basis of $\mathfrak{s}_2$ is given by

$$\partial_t, \quad \partial_x, \quad \partial_y, \quad t\partial_x + \partial_u, \quad t\partial_y + \partial_v. \tag{4.19}$$

As in the previous Section 4.4 we could additionally include the scaling symmetries of the equations (4.1) in the subalgebra $\mathfrak{s}_2$, referring to them as equivalence transformations of the class of doubly periodic boundary value problems. The reason why we did not include these scalings above is because all the discretizations for the shallow-water equations we use subsequently do not change the scaling properties of that system. This means that these discretizations satisfy the required scaling properties already by construction. On the other hand, the additional presence of scaling operators would slightly complicate the expressions for the difference invariants

computed below, without giving any significant new information (the additional arising coefficients will factor out anyway for the resulting schemes). Only in the course of setting up the invariant grid generator, it will be necessary to explicitly take into account the specific scaling symmetries, which will consistently be done in Section 4.5.3. Note that both symmetries (4.19) and scaling symmetries of the two-dimensional shallow-water equations (4.1) generate equivalence transformations of the set of relevant initial conditions.

As can be envisioned from the consideration of the numerical models of the one-dimensional shallow-water equations discussed in Section 4.4, discretization schemes for the two-dimensional shallow-water equations will be invariant under Galilean symmetries only if they are based on adaptive grids. As for the channel model Galilean transformations are only admitted in the x-direction, it suffices for a grid to be adaptive in the x-direction. This in particular means that we can use a uniform spacing in the y-direction and have a spatial grid with changing resolution only along the channel. On the other hand, the shallow-water equations with double periodic boundary conditions require the treatment of adaptive grids in both the x- and y-direction. An initial orthogonal spatial grid is driven to a non-orthogonal grid, which makes the direct evaluation of finite difference derivatives much more elaborate. This problem is treated upon using a finite volume formulation of that scheme.

For simplicity, all the schemes are developed on an Arakawa A-grid subsequently, i.e. the variables u, v, h are defined in the same respective points. See e.g. [42] for a discussion of different types of staggered grids for the shallow-water equations. The usage of other types of grid staggering can be done in a similar way as that shown for the A-grid subsequently.

4.5.2 Invariant numerical schemes with double periodic boundary conditions: Lagrangian scheme

The main difficulty with adaptive grids in both the x- and y-direction is that it can become cumbersome to directly evaluate the gradients of the dependent variables on such curvilinear grids by finite differences. As discussed in Section 4.4, a prominent strategy to overcome this difficulty is to introduce a mapping from the computational (logical) coordinates (ξ, η) to the physical coordinates (x, y). This will be done in the subsequent Section 4.5.3.

For the sake of demonstration we take another, more direct approach here, namely using the finite volume formulation of the divergence operator, see e.g. [44]. Using the theorem of Gauß–Ostrogradsky, we can approximate the divergence $\nabla \cdot \mathbf{f}$ of a vector-function $\mathbf{f}$ over a single grid cell with area A and edge lengths l_i as

$$\nabla \cdot \mathbf{f} \approx \frac{1}{A} \sum_{i=1}^{4} (\mathbf{f}_i \cdot \mathbf{n}_i) l_i.$$

In the above formula, $\mathbf{n}_i$ denotes the outward directed unit vector at the single cell edges.

As it is possible to cast the shallow-water equations (4.1) into conserved (momentum) form

$$\begin{aligned} \mathrm{Eq}^h &= h_t + (hu)_x + (hv)_y = 0, \\ \mathrm{Eq}^u &= (hu)_t + \left(hu^2 + \frac{1}{2} h^2 \right)_x + (huv)_y = 0, \\ \mathrm{Eq}^v &= (hv)_t + (huv)_x + \left(hv^2 + \frac{1}{2} h^2 \right)_y = 0, \end{aligned} \tag{4.20}$$

the above approximation of the divergence operator is sufficient to discretize the two-dimensional shallow-water equations using the finite volume form.[1] On the other hand, a finite volume discretization is readily applicable on adaptive grids, as it is not necessary to approximate derivatives by finite differences in such a formulation.

So as to discretize (4.20) in an invariant way using the Dorodnitsyn method we would need a set of difference invariants and construct the discretization using these invariants as building blocks for the numerical scheme. The problem with this approach is the same as reported in the one-dimensional case. The Galilean transformation $\tilde{t} = t$, $\tilde{x} = x + \varepsilon_1 t$, $\tilde{y} = y + \varepsilon_2 t$, $\tilde{h} = h$, $\tilde{u} = u + \varepsilon_1$, $\tilde{v} = v + \varepsilon_2$ maps the system (4.20) to

$$\widetilde{\mathrm{Eq}}^h = \mathrm{Eq}^h, \quad \widetilde{\mathrm{Eq}}^u = \mathrm{Eq}^u + \varepsilon_1 \mathrm{Eq}^h, \quad \widetilde{\mathrm{Eq}}^v = \mathrm{Eq}^v + \varepsilon_2 \mathrm{Eq}^h, \tag{4.21}$$

i.e. it leads to a combination of the momentum equations with the continuity equation. Expressing the momentum equations in terms of differential invariants thus again only works by combining these equations with the continuity equation. As in the one-dimensional case, it is therefore not natural to attempt finding an invariant approximation of the momentum form of the shallow-water equations using difference invariants.

At the same time, the Lagrangian form of the shallow-water equations (4.1), which is

$$\frac{\mathrm{d}x}{\mathrm{d}t} = u, \quad \frac{\mathrm{d}y}{\mathrm{d}t} = v, \quad \frac{\mathrm{d}h}{\mathrm{d}t} + h\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0, \quad \frac{\mathrm{d}u}{\mathrm{d}t} + \frac{\partial h}{\partial x} = 0, \quad \frac{\mathrm{d}v}{\mathrm{d}t} + \frac{\partial h}{\partial y} = 0, \tag{4.22}$$

can be approximated using the finite volume method as well. Expressing an invariant discretization of (4.22) in terms of difference invariants is considerably easier than doing the same for an invariant discretization of (4.20). As in the one-dimensional case, the drawback of using (4.22) as a starting point is that the resulting scheme does not approximate a conserved form and thus neither preserves mass, momenta nor energy.

The stencil of the discretizations we aim to use is given in Fig. 4.4. All the dependent variables are defined in the centroids of the respective polygons. The fluxes through the edges will govern the evolution of these centroid values. In order to facilitate the computation of the fluxes it is necessary to determine the values of $w = (u, v, h)$ in the cell corners, which is done by interpolation. While in principle any type of interpolation can be used, we employ natural neighbors interpolation for this purpose, i.e. the values at the cell corners are $w_j = \sum_{\kappa=1}^{4} \rho^{\kappa,j} w_0^{\kappa,j}$, where $j = 1, \dots, 4$ and $w_0^{\kappa,j}$ are the values of w in the centers of those cells having in common the corner denoted by j. The interpolation weights $\rho^{\kappa,j}$ are determined in the following way. The Voronoi tessellation generated by the cell centers is constructed. Then a new tessellation is computed in which the point (x_j, y_j) is introduced as an additional generator. Let denote by $A_{P_0^\kappa}$ the area of the Voronoi cell of the original tessellation associated with the center point $P_0^\kappa = (x_0^\kappa, y_0^\kappa)$ and by A_{P_j} the area of the new cell associated with the corner point $P_j = (x_j, y_j)$ introduced for the second tessellation. Then the weights $\rho^{\kappa,j}$ are computed as $\rho^{\kappa,j} = (A_{P_j} \cap A_{P_0^\kappa})/A_{P_j}$. Once the values w_j are obtained, they can be regarded as proper stencil variables.

The prolongations of the symmetry operators (4.19) on the variables of the stencil shown in

[1] For equations including curl-terms, they can be converted into finite volume representation using the Stokes theorem.

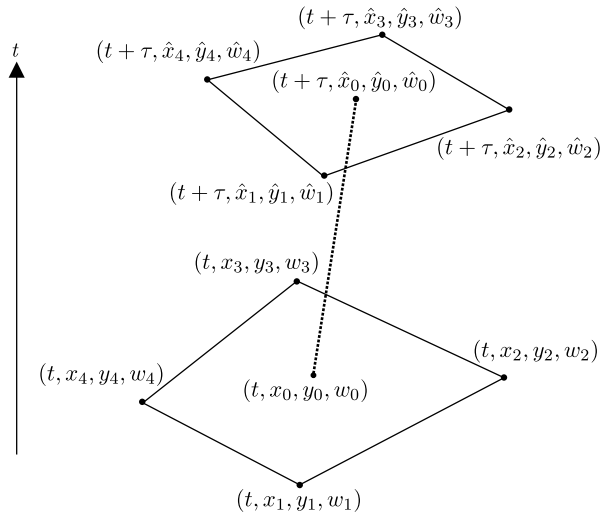

Figure 4.4: Stencil for the invariant Lagrangian schemes for the two-dimensional shallow-water equations with double periodic boundary conditions. The dependent variables $w = (u, v, h)$ are defined in the center (x_0, y_0) of the cells of the respective polygons. The fluxes are computed using the values at the corners $(x_k, y_k), k = 1 \dots 4$ of the cells, which are obtained by interpolation from the values at the polygon centroids. Variables with hat are those at the subsequent time step.

Fig. 4.4 read

$$\begin{aligned}
&\partial_t, \quad \sum_{i=0}^{4}(\partial_{x_i} + \partial_{\hat{x}_i}), \quad \sum_{i=0}^{4}(\partial_{y_i} + \partial_{\hat{y}_i}),\\
&\sum_{i=0}^{4}(t\partial_{x_i} + (t+\tau)\partial_{\hat{x}_i} + \partial_{u_i} + \partial_{\hat{u}_i}), \quad \sum_{i=0}^{4}(t\partial_{y_i} + (t+\tau)\partial_{\hat{y}_i} + \partial_{v_i} + \partial_{\hat{v}_i}).
\end{aligned} \tag{4.23}$$

These prolongations are well agreed with the above interpolation procedure. The difference invariants of the set (4.23) are given by

$$\begin{aligned}
&\tau, \quad h_i, \quad \hat{h}_i, \quad x_i - x_j, \quad y_i - y_j, \quad \hat{x}_i - x_j - \tau u_k, \quad \hat{y}_i - y_j - \tau v_k,\\
&u_i - u_j, \quad \hat{u}_i - u_j, \quad v_i - v_j, \quad \hat{v}_i - v_j,
\end{aligned}$$

where the indices i, j and k take the values $0, \dots, 4$. Note that of course not all of the above difference invariants are independent if i, j and k separately run through all possible values.

A simple explicit invariant scheme (Euler forward scheme) that can be constructed using these invariants is

$$\begin{aligned}
&\frac{\hat{x}_0 - x_0}{\tau} - u_0 = 0, \quad \frac{\hat{y}_0 - y_0}{\tau} - v_0 = 0,\\
&\frac{\hat{h}_0 - h_0}{\tau} + \frac{h_0}{2A}\sum_{i=1}^{4}[(u_i + u_{i+1})(y_{i+1} - y_i) - (v_i + v_{i+1})(x_{i+1} - x_i)] = 0,\\
&\frac{\hat{u}_0 - u_0}{\tau} + \frac{1}{2A}\sum_{i=1}^{4}(h_i + h_{i+1})(y_{i+1} - y_i) = 0,\\
&\frac{\hat{v}_0 - v_0}{\tau} - \frac{1}{2A}\sum_{i=1}^{4}(h_i + h_{i+1})(x_{i+1} - x_i) = 0,
\end{aligned} \tag{4.24}$$

where $A = \frac{1}{2}\sum_{i=1}^{4}(x_i y_{i+1} - x_{i+1}y_i)$ is the area of the polygon spanned by $(x_1, y_1), \ldots, (x_4, y_4)$ and $(x_5, y_5, u_5, v_5, h_5) = (x_1, y_1, u_1, v_1, h_1)$ by definition. As in the one-dimensional case, in the continuous limit this scheme converges to the Lagrangian representation of the two-dimensional shallow-water equations (4.22).

In a similar manner, we can formulate the implicit scheme (trapezoidal rule)

$$
\begin{aligned}
&\frac{\hat{x}_0 - x_0}{\tau} - \frac{1}{2}(u_0 + \hat{u}_0) = 0, \quad \frac{\hat{y}_0 - y_0}{\tau} - \frac{1}{2}(v_0 + \hat{v}_0) = 0,\\
&\frac{\hat{h}_0 - h_0}{\tau} + \frac{h_0}{4A}\sum_{i=1}^{4}[(u_i + u_{i+1})(y_{i+1} - y_i) - (v_i + v_{i+1})(x_{i+1} - x_i)] +\\
&\qquad\qquad \frac{\hat{h}_0}{4\hat{A}}\sum_{i=1}^{4}[(\hat{u}_i + \hat{u}_{i+1})(\hat{y}_{i+1} - \hat{y}_i) - (\hat{v}_i + \hat{v}_{i+1})(\hat{x}_{i+1} - \hat{x}_i)] = 0,\\
&\frac{\hat{u}_0 - u_0}{\tau} + \frac{1}{4A}\sum_{i=1}^{4}(h_i + h_{i+1})(y_{i+1} - y_i) + \frac{1}{4\hat{A}}\sum_{i=1}^{4}(\hat{h}_i + \hat{h}_{i+1})(\hat{y}_{i+1} - \hat{y}_i) = 0,\\
&\frac{\hat{v}_0 - v_0}{\tau} - \frac{1}{4A}\sum_{i=1}^{4}(h_i + h_{i+1})(x_{i+1} - x_i) - \frac{1}{4\hat{A}}\sum_{i=1}^{4}(\hat{h}_i + \hat{h}_{i+1})(\hat{x}_{i+1} - \hat{x}_i) = 0.
\end{aligned}
\tag{4.25}
$$

Fig. 4.5 shows the result of a numerical integration with the scheme (4.25) supplemented with periodic boundary conditions and specific initial conditions. The numerical solution of the water height h at $t = 2$ is shown in the left panel. The right panel depicts the associated discretization grid at $t = 2$. As in the case of the one-dimensional Lagrangian scheme, a strong distortion of the grid cells is visible, which is not directly related to pronounced features in the numerical solution, but rather a consequence of the Lagrangian grid movement. As both the discretizations (4.24) and (4.25) do not approximate the conserved form of the shallow-water equations (4.20), they neither conserve the mass $\mathcal{M}$ and the momenta $\mathcal{P}_x$, $\mathcal{P}_y$ nor the energy $\mathcal{H}$.

It should be stressed that it is possible to formulate an invariant finite volume scheme for the conserved form of the shallow-water equations (4.20), similar as was shown in the previous section for the one-dimensional case. As said above, the problem of doing this systematically within the Dorodnitsyn approach is that it can be hard to find a proper combination of elementary difference invariants that allows one to approximate the momentum form of the shallow-water equations. This is why we will show an alternative way of constructing invariant numerical schemes for the two-dimensional shallow-water equations in the following section, which will avoid the technical complications that can arise when using the difference invariants method for the construction of symmetry-preserving numerical schemes.

4.5.3 Invariant numerical schemes with double periodic boundary conditions: Eulerian scheme

Though the finite volume discretization developed in Section 4.5.2 is suitable from the point of view of invariance preservation, it is not ideal from the viewpoint of numerical analysis. In general, Lagrangian schemes are not in widespread use as they can easily lead to tangling meshes or rapidly changing grids through the spatial domain. For the same reason, numerical schemes in hydrodynamics are usually formulated in terms of Eulerian variables (or using some combination of Eulerian and Lagrangian schemes). An invariant scheme on an adaptive grid

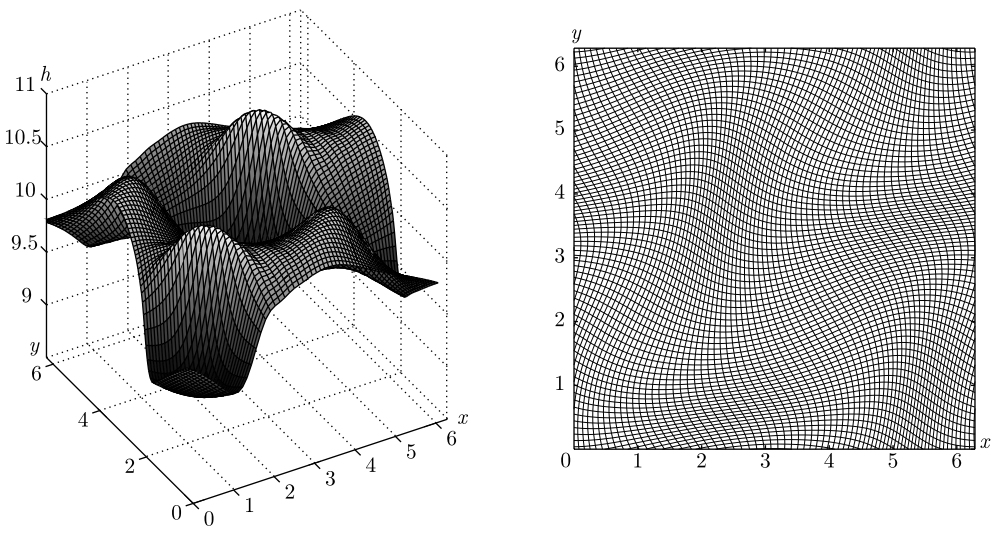

Figure 4.5: Numerical integration of the two-dimensional shallow-water equations (4.1) using the scheme (4.25) with $\tau = 0.001$ and $N_x \times N_y = 71 \times 71$ grid points on the square $[0, 2\pi] \times [0, 2\pi]$ over the time interval $[0, 2]$. The initial conditions are $u = A \sin(x + \varphi_0) \sin y$, $v = A \sin x \sin y$ and $h = h_0 + A \cos(x + \varphi_0) \cos y$, with $A = 0.4$, $\varphi_0 = \pi/6$ and $h_0 = 10$. Left: Numerical solution for h at $t = 2$. Right: Spatial discretization grid at $t = 2$.

can be formulated by combining the idea of having an invariant grid generator proposed in Section 4.4.3 with the discretization in computational coordinates. More specifically, we consider the momentum form (4.20) of the two-dimensional shallow-water equations and re-write it in the computational coordinates $\theta = t$, $\xi = \xi(t, x, y)$, $\eta = \eta(t, x, y)$:

$$\frac{\partial}{\partial\theta}(JF^t) + \frac{\partial}{\partial\xi}(J\xi_t F^t + J\xi_x F^x + J\xi_y F^y) + \frac{\partial}{\partial\eta}(J\eta_t F^t + J\eta_x F^x + J\eta_y F^y) = 0, \tag{4.26}$$

where $F^t = (h, hu, hv)$, $F^x = (hu, hu^2 + \frac{1}{2}h^2, huv)$, $F^y = (hv, huv, hv^2 + \frac{1}{2}h^2)$ and

$$J = x_\xi y_\eta - x_\eta y_\xi, \quad \xi_t = -\xi_x x_\theta - \xi_y y_\theta, \quad \eta_t = -\eta_x x_\theta - \eta_y y_\theta,$$
$$\xi_x = \frac{y_\eta}{J}, \quad \xi_y = -\frac{x_\eta}{J}, \quad \eta_x = -\frac{y_\xi}{J}, \quad \eta_y = \frac{x_\xi}{J}.$$

The invariance of the system (4.26) with respect to shifts of the former independent variables t, x and y is obvious since the left-hand sides of equations of the system (which we denote by Eq^h, Eq^u and Eq^v, respectively) do not explicitly involve these variables. Therefore, any finite difference approximation of (4.26) is invariant with respect to the above shifts extended to the corresponding stencil, cf. (4.23). Note that any involved transformation is trivially extended to the computational coordinates ξ and η, i.e., they are not transformed. The scale symmetry transformations of the shallow-water equations (4.1) are automatically preserved in the course of a proper finite difference approximation of (4.26), see the related discussion in Section 4.5.1. In order to make clear the invariance with respect to Galilean boosts, we recombine terms in (4.26) substituting $J\xi_t = -J\xi_x x_\theta - J\xi_y y_\theta$ and $J\eta_t = -J\eta_x x_\theta - J\eta_y y_\theta$:

$$\frac{\partial}{\partial\theta}(JF^t) + \frac{\partial}{\partial\xi}\big(J\xi_x(F^x - x_\theta F^t) + J\xi_y(F^y - y_\theta F^t)\big) +$$
$$\frac{\partial}{\partial\eta}\big(J\eta_x(F^x - x_\theta F^t) + J\eta_y(F^y - y_\theta F^t)\big) = 0,$$

The Galilean boost $\tilde{t} = t$, $\tilde{x} = x + \varepsilon_1 t$, $\tilde{y} = y + \varepsilon_2 t$, $\tilde{h} = h$, $\tilde{u} = u + \varepsilon_1$, $\tilde{v} = v + \varepsilon_2$ maps the system (4.26) to the system with

$$\widetilde{\mathrm{Eq}}^h = \mathrm{Eq}^h, \quad \widetilde{\mathrm{Eq}}^u = \mathrm{Eq}^u + \varepsilon_1 \mathrm{Eq}^h, \quad \widetilde{\mathrm{Eq}}^v = \mathrm{Eq}^v + \varepsilon_2 \mathrm{Eq}^h.$$

Note that this is the same transformation law in computational variables as it is in the physical space (4.21). *The main idea for finding invariant numerical schemes of* (4.26) *is to construct the discretization in such a manner that the discrete counterpart of the system* (4.26) *is transformed similarly by the extension of the Galilean boost to the stencil points.* In order to preserve Galilean boosts as symmetries in the course of discretization, it suffices

- to use the same discretization schemes for all the equations of the system (4.26), just varying the number of the corresponding components of F^t, F^x and F^y;
- to evaluate all the copies of $J\xi_x$ (resp. the components F^t and F^x) related to the block $J\xi_x(F^x - x_\theta F^t)$ in the same grid point and in the same way; the same rule should be applied for the other similar blocks, $J\xi_y(F^y - y_\theta F^t)$, $J\eta_x(F^x - x_\theta F^t)$ and $J\eta_y(F^y - y_\theta F^t)$.

For example, consider the trapezoidal scheme for the system (4.26)

$$\frac{\hat{J}_{jk}\hat{F}^t_{jk} - J_{jk}F^t_{jk}}{\tau} + \frac{U_{jk} + \hat{U}_{jk}}{2} + \frac{V_{jk} + \hat{V}_{jk}}{2} = 0, \tag{4.27}$$

where τ is the step in $\theta = t$,

$$\begin{aligned}
U_{jk} = \frac{1}{2\Delta\xi}[&(J\xi_t)_{j+1/2,k}(F^t_{jk} + F^t_{j+1,k}) - (J\xi_t)_{j-1/2,k}(F^t_{jk} + F^t_{j-1,k}) + \\
&(J\xi_x)_{j+1/2,k}(F^x_{jk} + F^x_{j+1,k}) - (J\xi_x)_{j-1/2,k}(F^x_{jk} + F^x_{j-1,k}) + \\
&(J\xi_y)_{j+1/2,k}(F^y_{jk} + F^y_{j+1,k}) - (J\xi_y)_{j-1/2,k}(F^y_{jk} + F^y_{j-1,k})], \\
V_{jk} = \frac{1}{2\Delta\eta}[&(J\eta_t)_{j,k+1/2}(F^t_{jk} + F^t_{j,k+1}) - (J\eta_t)_{j,k-1/2}(F^t_{jk} + F^t_{j,k-1}) + \\
&(J\eta_x)_{j,k+1/2}(F^x_{jk} + F^x_{j,k+1}) - (J\eta_x)_{j,k-1/2}(F^x_{jk} + F^x_{j,k-1}) + \\
&(J\eta_y)_{j,k+1/2}(F^y_{jk} + F^y_{j,k+1}) - (J\eta_y)_{j,k-1/2}(F^y_{jk} + F^y_{j,k-1})],
\end{aligned}$$

the values J, $J\xi_x = y_\eta$, $J\xi_y = -x_\eta$, $J\eta_x = -y_\xi$, $J\eta_y = x_\xi$, $J\xi_t$ and $J\eta_t$ are discretized in the following way:

$$\begin{aligned}
J_{jk} &= \frac{1}{4\Delta\xi\Delta\eta}[(x_{j+1,k} - x_{j-1,k})(y_{j,k+1} - y_{j,k-1}) - (x_{j,k+1} - x_{j,k-1})(y_{j+1,k} - y_{j-1,k})], \\
(J\xi_x)_{j\pm1/2,k} &= \frac{1}{4\Delta\eta}(y_{j,k+1} - y_{j,k-1} + y_{j\pm1,k+1} - y_{j\pm1,k-1}), \\
(J\xi_y)_{j\pm1/2,k} &= -\frac{1}{4\Delta\eta}(x_{j,k+1} - x_{j,k-1} + x_{j\pm1,k+1} - x_{j\pm1,k-1}), \\
(J\eta_x)_{j,k\pm1/2} &= -\frac{1}{4\Delta\xi}(y_{j+1,k} - y_{j-1,k} + y_{j+1,k\pm1} - y_{j-1,k\pm1}), \\
(J\eta_y)_{j,k\pm1/2} &= \frac{1}{4\Delta\xi}(x_{j+1,k} - x_{j-1,k} + x_{j+1,k\pm1} - x_{j-1,k\pm1}), \\
(J\xi_t)_{j\pm1/2,k} &= -(J\xi_x)_{j\pm1/2,k}\frac{\dot{x}_{jk} + \dot{x}_{j\pm1,k}}{2} - (J\xi_y)_{j\pm1/2,k}\frac{\dot{y}_{jk} + \dot{y}_{j\pm1,k}}{2}, \\
(J\eta_t)_{j,k\pm1/2} &= -(J\eta_x)_{j,k\pm1/2}\frac{\dot{x}_{jk} + \dot{x}_{j,k\pm1}}{2} - (J\eta_y)_{j,k\pm1/2}\frac{\dot{y}_{jk} + \dot{y}_{j,k\pm1}}{2},
\end{aligned}$$

where $\dot{x}_{jk} = (\hat{x}_{jk} - x_{jk})/\tau$ and $\dot{y}_{jk} = (\hat{y}_{jk} - y_{jk})/\tau$ are by definition the mesh velocities in the x- and y-direction, respectively, and by hat we mark the corresponding values at the time $\theta + \tau$. In particular, we take

$$(\hat{J\xi_t})_{j\pm1/2,k} = -(\hat{J\xi_x})_{j\pm1/2,k}\frac{\dot{x}_{jk} + \dot{x}_{j\pm1,k}}{2} - (\hat{J\xi_y})_{j\pm1/2,k}\frac{\dot{y}_{jk} + \dot{y}_{j\pm1,k}}{2},$$
$$(\hat{J\eta_t})_{j,k\pm1/2} = -(\hat{J\eta_x})_{j,k\pm1/2}\frac{\dot{x}_{jk} + \dot{x}_{j,k\pm1}}{2} - (\hat{J\eta_y})_{j,k\pm1/2}\frac{\dot{y}_{jk} + \dot{y}_{j,k\pm1}}{2}.$$

As the system of difference equations (4.27) satisfies the above conditions, it is invariant with respect to properly extended Galilean boosts.

Remark 4.3. The usage of computational coordinates also enlightens the subtle change of the meaning of the time derivatives in a number of papers devoted to the construction of invariant numerical schemes, such as in [21, 52]. While in the standard (Eulerian) discretization, the continuous limit of the form $(\hat{u} - u)/\tau$ yields the partial derivative u_t, in the framework of invariant schemes these terms are often to be interpreted as Lagrangian time derivatives $\dot{u}$ (see also Section 4.4.2). This immediate transition from an Eulerian (partial) time derivative to a Lagrangian (total) time derivative is a necessary consequence of the intermediate step of discretizing an equation in computational coordinates assuming that the grid evolution is described by the equations $\dot{x} = u$, and $\dot{y} = v$, see also the discussion in [43].

Remark 4.4. It should be noted that computational coordinates have a clear physical meaning in the present context. As they do not change during the evolution of the grid, they can be interpreted as the Lagrangian variables (fluid labels) of fluid mechanics provided we again assume a Lagrangian grid evolution. A prominent way to choose these fluid labels is by setting them to equal the Cartesian coordinates at the onset of evolution. By definition, this is the same role that computational coordinates play in the numerics of moving meshes. Stated in another way, the requirement of maintaining invariance of the discretization scheme and discretization stencil of the shallow-water equations under the Galilean group naturally boils down to discretize these equations in computational coordinates.

It then remains to specify the grid velocities $\dot{x}$ and $\dot{y}$ in order to complete the scheme given in (4.27). This can be done in a similar manner as in Section 4.4.3 using the idea of equidistributing meshes (though, strictly speaking, equidistribution in higher dimensions is not sufficient to uniquely determine an adaptive grid, see e.g. the discussion in [11]). Thus, the grid will be determined from the system of elliptic equations

$$\nabla_{\boldsymbol{\xi}} \cdot (\mathbb{G}\nabla_{\boldsymbol{\xi}} x) = 0, \quad \nabla_{\boldsymbol{\xi}} \cdot (\mathbb{G}\nabla_{\boldsymbol{\xi}} y) = 0, \tag{4.28}$$

where $\nabla_{\boldsymbol{\xi}}$ denotes the gradient in the space of computational coordinates (ξ, η), $\mathbb{G} = w\mathbb{I}$ is the matrix-valued monitor function, for $\mathbb{I}$ being the two-by-two unit matrix and $w = w(x, y)$ being a weight function which depends on the (numerical) solution of the shallow-water equations [18]. An invariant discretization of the first equation reads

$$\frac{1}{\Delta\xi}\left(w_{i+1/2,j}\frac{\hat{x}_{i+1,j} - \hat{x}_{ij}}{\Delta\xi} - w_{i-1/2,j}\frac{\hat{x}_{ij} - \hat{x}_{i-1,j}}{\Delta\xi}\right) +$$
$$\frac{1}{\Delta\eta}\left(w_{i,j+1/2}\frac{\hat{x}_{i,j+1} - \hat{x}_{ij}}{\Delta\eta} - w_{i,j-1/2}\frac{\hat{x}_{ij} - \hat{x}_{i,j-1}}{\Delta\eta}\right) = 0,$$

where

$$w_{i+1/2,j}=\frac{w_{i+1,j}+w_{ij}}{2},\quad w_{i-1/2,j}=\frac{w_{ij}+w_{i-1,j}}{2},$$
$$w_{i,j+1/2}=\frac{w_{i,j+1}+w_{ij}}{2},\quad w_{i,j-1/2}=\frac{w_{ij}+w_{i,j-1}}{2}$$

and similar for the second equation, provided that w is approximated by a difference invariant of the algebra $\mathfrak{s}_2$, which is spanned by the vector fields (4.19). Once again, straightforward choices for w that can be discretized using difference invariants are

$$w^1=\sqrt{1+\alpha(u_x^2+u_y^2+v_x^2+v_y^2)},\qquad w^2=\sqrt{1+\alpha(h_{xx}+h_{yy})^2},$$

but of course other forms for w are possible as well. Similarly to the one-dimensional case, cf. Remark 4.1, the general form of w which is an invariant of the algebra $\mathfrak{s}_2$ is given by an arbitrary smooth function of derivatives of u, v and h with respect to x and y including h itself but not u and v. At the same time, we should also take into accout other desired properties of w. Both the functions w^1 and w^2 are invariant with respect to shifts and Galilean boosts generated by vector fields (4.19), the scalings generated by the vector field $x\partial_x+y\partial_y+u\partial_u+v\partial_v+2h\partial_h$ and even rotations. All the scaling symmetries of the shallow-water equations are at least equivalence transformations for the sets of functions of such forms, where the parameter α is varied.

It should also be stressed that the grid generator based on system (4.28) is a rather simple one. More advanced formulations of grid generators exist, e.g. by using a general positive definite and symmetric matrix $\mathbb{G}$. Alternatively, the grids at a certain time level could be computed using moving mesh partial differential equations [8, 11], provided it would be possible to discretize such equations in an invariant way.

A different methodology is to use so-called *velocity based* moving mesh strategies. Unlike the *location based* methods, which were exclusively used in the present paper, then not the location of the grid points is determined directly but rather equations for the mesh velocity are formulated. Velocity based strategies, such as the method involving the geometric conservation law [11, 17], provide alternative ways of formulating grid equations that give a basis to realize invariant moving mesh equations. See also [48], where the term *geometric conservation law* was introduced.

In Fig. 4.6 we repeat the numerical integration of the two-dimensional shallow-water equations with the setting of Fig. 4.5 but now using the scheme (4.27) in combination with a grid generator based on w^2. Similarly to the case of the one-dimensional shallow-water equations it can be seen from Fig. 4.6 that the usage of a grid generator leads to grids that are not as distorted as those obtained from a purely Lagrangian scheme. Moreover, the regions of grid concentration are now directly linked to the physical behavior of the numerical solution for the dependent variable h. The scheme (4.27) is mass and momenta conserving but, as all the other schemes presented in the paper, dissipates the energy. The conserved quantities are evaluated at time t as

$$\mathcal{M}=\Delta\xi\Delta\eta\sum_{j,k}h_{jk}J_{jk},\quad \mathcal{P}_x=\Delta\xi\Delta\eta\sum_{j,k}h_{jk}u_{jk}J_{jk},$$
$$\mathcal{P}_y=\Delta\xi\Delta\eta\sum_{j,k}h_{jk}v_{jk}J_{jk},\quad \mathcal{H}=\frac{1}{2}\Delta\xi\Delta\eta\sum_{j,k}(h_{jk}(u_{jk}^2+v_{jk}^2)+h_{jk}^2)J_{jk}.$$

4.6 Conclusion

The present paper is devoted to the construction of several invariant numerical schemes modeling shallow-water dynamics. In particular, we aim to describe a possible bridge between the

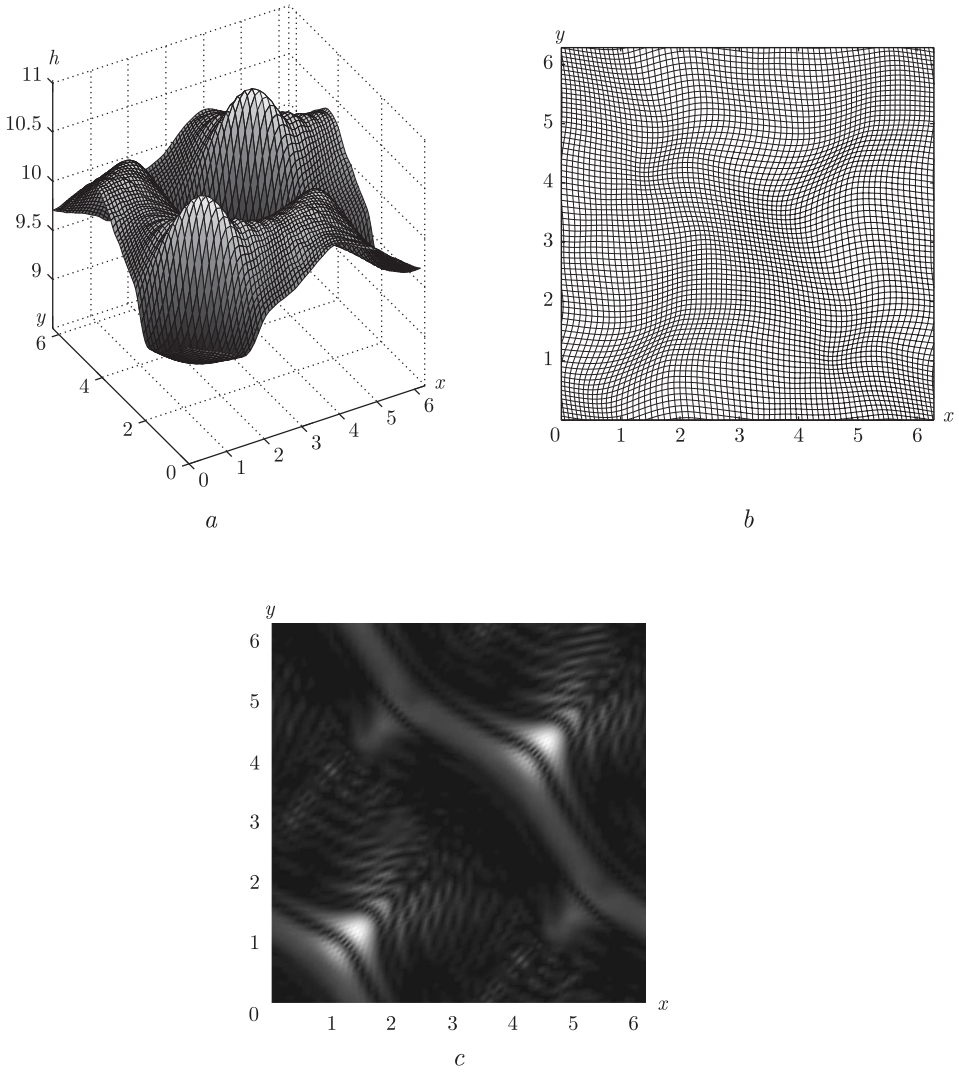

Figure 4.6: Numerical integration of the two-dimensional shallow-water equations (4.1) using the scheme (4.27) with $\tau = 0.001$ and $N_x \times N_y = 71 \times 71$ grid points on the square $[0, 2\pi] \times [0, 2\pi]$ over the time interval $[0, 2]$. The initial conditions are $u = A \sin(x + \varphi_0) \sin y$, $v = A \sin x \sin y$ and $h = h_0 + A \cos(x + \varphi_0) \cos y$, with $A = 0.4$, $\varphi_0 = \pi/6$ and $h_0 = 10$. As a weight function, $w^2 = \sqrt{1 + \alpha(h_{xx} + h_{yy})^2}$ is chosen with $\alpha = 0.4$. (a) Numerical solution for h at $t = 2$. (b) Spatial discretization grid at $t = 2$. (c) The weight function at $t = 2$.

formalism of constructing invariant numerical schemes and the existing theory on adaptive moving meshes. Such a bridge was already indicated in the literature. Indeed, there already exist a number of investigations devoted to the importance of scale invariance in the theory of moving mesh equations. Thus, in [7, 8, 10] (see also [11, p. 111] and references therein) a moving mesh partial differential equation was constructed that preserves the scale invariance of the physical differential equation to be discretized. The extension of this idea to setting up a grid generator that is invariant with respect to (a suitable subgroup of) the maximal Lie invariance group of a system of differential equations is therefore straightforward and was conceptually indicated in the aforementioned sources. The idea of introducing computational coordinates into invariant numerical schemes has also been successfully demonstrated for one-dimensional nonlinear Schrödinger equations [8].

We require our discretization schemes to be invariant with respect to the subgroup of the maximal Lie symmetry group of the (resp. one- or two-dimensional) shallow-water equations admitted when imposing periodic boundary conditions. From the physical point of view it is natural to assume that appropriate symmetries of the system of differential equations under considerations act as equivalence transformations on a joint class of physically relevant boundary value problems. Imposing periodic boundary conditions for varying intervals in the one-dimensional case (resp. for rectangular domains of varying sizes whose sides are parallel to coordinate axes in the two-dimensional case) while both the initial time and initial conditions also vary, we select the subgroup generated by the time and space translations, the Galilean boosts and the scalings symmetries of the shallow-water equations. Other subgroups might be chosen as well, but for wide or even infinite-dimensional maximal Lie invariance (pseudo)groups admitted by the prominent models in hydrodynamics it can be quite intricate to justify the choice for such subgroups from the physical point of view.

In general, the inclusion of well-proven principles in the study of invariant numerical schemes is a task requested. The invariant schemes for numerous evolution equations constructed so far were mostly purely Lagrangian schemes. However, these schemes are not in prevalent use in practice as they usually lead to complicated mesh geometries which might eventually (at least locally) degrade the quality of the numerical solution. This can be seen directly by comparing Figures 4.5 and 4.6, where the stronger distortion of the grid lines in the Lagrangian scheme is manifest already after a relatively short period of integration. Therefore, the formulation of invariant grid generators coupled with suitable invariant discrete counterparts of physical systems of differential equations is a practicable way for symmetry preserving numerical integration of these systems.

A further novel feature of the present paper is the construction of invariant numerical schemes for higher-dimensional systems of partial differential equations. Higher-dimensional schemes are especially challenging if it is not possible to use fixed orthogonal grids. In the course of constructing such schemes for the two-dimensional shallow-water equations we have shown that invariant discretizations are not only restricted to finite difference schemes. It is possible and straightforward to also formulate finite volume discretizations that preserve symmetries of systems of differential equations. In a similar manner, other discretization techniques, such as the finite element method, could be employed as well.

This problem can be tackled by transforming the system under consideration into computational coordinates. We have shown that the transition to these coordinates is a natural step in the course of the construction of Galilean invariant discrete schemes. The key to the construction is then not to simply combine difference invariants as proposed in the original method by Dorodnitsyn but to study the transformation laws of the equations in computational coordinates for the respective symmetries. These laws are trivial for all the admitted symmetries except for the Galilean boosts. For Galilean invariance it is found that the new momentum equations are given as the combination of the old momentum and continuity equations. An invariant discretization is therefore achieved by finding proper discrete counterparts of these transformation laws rather than combining difference invariants.

Because the main objective of this paper is to demonstrate different strategies for finding invariant discretization schemes exemplified with the shallow-water equations, the discretization schemes considered are kept as simple as possible. This concerns both the design of the schemes themselves (e.g. using only two-level time integration methods and unstaggered grids) as well as the solution of the resulting algebraic equations, which is done in the most direct and straight-

forward manner. More advanced integration and algebraic solution techniques can be readily adopted but their discussion is restrained to keep the focus of the paper on the conceptual aspect of introducing the Lie symmetry approach in the framework of numerical modeling as far as possible. For example, the extension to more advanced time integration methods such as arbitrary Runge–Kutta or general time-splitting schemes can be done by extension of the discretization stencils via inclusion of further time layers. Similarly, the usage of staggered grids can be facilitated by adding further points to the stencil on the same time layer. The procedure of invariant discretization involving wider stencils then follows precisely the same techniques as outlined and used in the present paper. Within the approach based on the construction of difference invariants, both ways of extending the stencils will lead to a larger number of invariants and thus to an increased number of possibilities for combining them to a particular discretization scheme.

It was mentioned in the introduction that a system of differential equations might possess various qualitative properties that one should aim to preserve in the course of setting up a numerical model. Besides symmetries, it is of outstanding importance to monitor the behavior of conserved quantities possessed by the system under consideration. This is a problem of central importance in long-term integrations of such systems as a systematic failure in capturing conservation laws may lead to unrealistic numerical results (e.g. loss of mass or wrong turbulence spectra). Proper discretizations of the momentum form of the shallow-water equations conserve the mass and momenta exactly or to high order, but none of them is actually energy conserving. This should not come as a complete surprise as setting up energy conserving schemes for the shallow-water equations is a quite nontrivial problem , see e.g. the schemes proposed in [2, 32, 45]. The inclusion of additional conserved quantities in the construction of invariant discretization schemes will therefore be one of our future research topics.

From a more general point of view, the requirement of preserving symmetries in a discretization scheme might lead to a geometric justification for using adaptive meshes. Though there are several classes of physical problems (such as blow-ups) for which adaptive meshes are well suited, the usage of such meshes is not undisputed in the numerical analysis and geophysical fluid dynamics communities. The drawbacks of moving meshes, such as an additional level of complexity of the schemes or the computational overhead resulting from computing and storing the mesh points at each time level must be well opposed to their potential benefits in a case-by-case basis. The result that the numerical preservation of important structural property like symmetries automatically requires to use moving meshes can thus be seen as a geometric argument for allowing adaptive discretization grid for certain classes of physical differential equations. Moreover, the usage of a grid redistribution equation (or r-adaptivity) as advocated in the present paper is also most suitable because it can be efficiently implemented within the framework of parallel computing.

Acknowledgements

The authors thank Prof. Pavel Winternitz and Dr. Matthias Sommer for useful discussions. The remarks by the anonymous referees are much appreciated. This research was supported by the Austrian Science Fund (FWF), projects J3182–N13 (AB), P20632 and Y237 (ROP).

Bibliography

[1] Andreev V.K., Kaptsov O.V., Pukhnachov V.V. and Rodionov A.A., *Applications of group-theoretical methods in hydrodynamics*, Kluwer, Dordrecht, 1998.

[2] Arakawa A. and Lamb V.R., A potential enstrophy and energy conserving scheme for the shallow water equations, *Mon. Wea. Rev.* **109** (1981), 18–36.

[3] Bakirova M.I., Dorodnitsyn V.A. and Kozlov R.V., Symmetry-preserving discrete schemes for some heat transfer equations, *J. Phys. A* **30** (1997), 8139–8155.

[4] Bihlo A., *Symmetry methods in the atmospheric sciences*, Ph.D. thesis, University of Vienna, 2010.

[5] Bihlo A. and Popovych R.O., Lie symmetries and exact solutions of the barotropic vorticity equation, *J. Math. Phys.* **50** (2009), 123102 (12 pages), arXiv:0902.4099.

[6] Bihlo A. and Popovych R.O., Lie symmetry analysis and exact solutions of the quasi-geostrophic two-layer problem, *J. Math. Phys.* **52** (2011), 033103 (24 pages), arXiv:1010.1542.

[7] Bluman G. and Kumei S., On invariance properties of the wave equation, *J. Math. Phys.* **28** (1987), 307–318.

[8] Bluman G. and Kumei S., *Symmetries and differential equations*, Springer, New York, 1989.

[9] Bluman G.W., Reid G.J. and Kumei S., New classes of symmetries for partial differential equations, *J. Math. Phys.* **29** (1988), 806–811.

[10] Bridges T.J. and Reich S., Numerical methods for Hamiltonian PDEs, *J. Phys. A* **39** (2006), 5287–5320.

[11] Budd C. and Dorodnitsyn V.A., Symmetry-adapted moving mesh schemes for the nonlinear Schrödinger equation, *J. Phys. A* **34** (2001), 10387–10400.

[12] Budd C.J. and Collins G.J., An invariant moving mesh scheme for the nonlinear diffusion equation, *Appl. Numer. Math.* **26** (1998), 23–39.

[13] Budd C.J. and Collins G.J., Symmetry based numerical methods for partial differential equations, in *Numerical analysis 1997 (Dundee)*, vol. 380 of *Pitman Res. Notes Math. Ser.*, Longman, Harlow, pp. 16–36, 1998.

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

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

[16] Budd C.J. and Iserles A., Geometric integration: numerical solution of differential equations on manifolds, *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **357** (1999), 945–956.

[17] Cao W., Huang W. and Russell R., A moving mesh method based on the geometric conservation law, *SIAM J. Sci. Comput.* **24** (2002), 118–142.

[18] Cao W., Huang W. and Russell R.D., A study of monitor functions for two-dimensional adaptive mesh generation, *SIAM J. Sci. Comput.* **20** (1999), 1978–1994.

[19] Chesnokov A.A., Symmetries and exact solutions of the rotating shallow-water equations, *Eur. J. Appl. Math.* **20** (2009), 461–477.

[20] Chhay M. and Hamdouni A., A new construction for invariant numerical schemes using moving frames, *C. R. Mecanique* **338** (2010), 97–101.

[21] Chhay M., Hoarau E., Hamdouni A. and Sagaut P., Comparison of some Lie-symmetry-based integrators, *J. Comp. Phys.* **230** (2011), 2174–2188.

[22] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[23] Dorodnitsyn V.A., Transformation groups in mesh spaces, *J. Sov. Math.* **55** (1991), 1490–1517.

[24] Dorodnitsyn V.A. and Kozlov R., A heat transfer with a source: the complete set of invariant difference schemes, *J. Nonlin. Math. Phys.* **10** (2003), 16–50.

[25] Dorodnitsyn V.A. and Winternitz P., Lie point symmetry preserving discretizations for variable coefficient Korteweg–de Vries equations, *Nonlin. Dyn.* **22** (2000), 49–59.

[26] Egger J., Volume conservation in phase space: a fresh look at numerical integration schemes, *Mon. Wea. Rev.* **124** (1996), 1955–1964.

[27] Fels M. and Olver P.J., Moving coframes: I. A practical algorithm, *Acta Appl. Math.* **51** (1998), 161–213.

[28] Frank J. and Reich S., The Hamiltonian particle–mesh method for the spherical shallow water equations, *Atmos. Sci. Lett.* **5** (2004), 89–95.

[29] Huang W. and Russell R.D., *Adaptive Moving Mesh Methods*, Springer, New York, 2010.

[30] Hydon P.E., *Symmetry methods for differential equations*, Cambridge University Press, Cambridge, 2000.

[31] Ibragimov N.H., Aksenov A.V., Baikov V.A., Chugunov V.A., Gazizov R.K. and Meshkov A.G., *CRC handbook of Lie group analysis of differential equations. Vol. 2. Applications in engineering and physical sciences. Edited by N. H. Ibragimov*, CRC Press, Boca Raton, 1995.

[32] Kim P., Invariantization of numerical schemes using moving frames, *BIT Numerical Mathematics* **47** (2007), 525–546.

[33] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Physica D* **237** (2008), 243–254.

[34] Leimkuhler B. and Reich S., *Simulating Hamiltonian dynamics*, Cambridge University Press, Cambridge, 2004.

[35] Morrison P.J. and Greene J.M., Noncanonical Hamiltonian density formulation of hydrodynamics and ideal magnetohydrodynamics, *Phys. Rev. Lett.* **45** (1980), 790–794.

[36] Olver P.J., Moving frames, *J. Symbolic Comput.* **36** (2003), 501–512.

[37] Olver P.J., Generating differential invariants, *J. Math. Anal. Appl.* **333** (2007), 450–471.

[38] Olver P.J., *Equivalence, invariants and symmetry*, Cambridge University Press, Cambridge, 2009.

[39] Ovsiannikov L.V., *Group analysis of differential equations*, Acad. Press, New York, 1982.

[40] Pavlenko A.S., Symmetries and solutions to equations of two-dimensional motions of polytropic gas, *Sib. Èlektron. Mat. Izv.* **2** (2005), 291–307.

[41] Pedlosky J., *Geophysical fluid dynamics*, Springer, New York, 1987.

[42] Randall D.A., Geostrophic adjustment and the finite-difference shallow-water equations, *Mon. Wea. Rev.* **122** (1994), 1371–1377.

[43] Reich S., The repulsive-particle method and the symplectic integration of the shallow-water equations, UniS preprint, 1999.

[44] Ringler T.D. and Randall D.A., A potential enstrophy and energy conserving numerical scheme for solution of the shallow-water equations on a geodesic grid, *Mon. Wea. Rev.* **130** (2002), 1397–1410.

[45] Salmon R., A general method for conserving energy and potential enstrophy in shallow-water models, *J. Atmos. Sci.* **64** (2007), 515–531.

[46] Sommer M. and Névir P., A conservative scheme for the shallow-water system on a staggered geodesic grid based on a Nambu representation, *Q. J. R. Meteorol. Soc.* **135** (2009), 485–494.

[47] Sommer M. and Reich S., Phase space volume conservation under space and time discretization schemes for the shallow-water equations, *Mon. Wea. Rev.* **138** (2010), 4229–4236.

[48] Thomas P.D. and Lombard C.K., Geometric conservation law and its applications to flow computations on moving grids, *AIAA Journal* **17** (1979), 1030–1037.

[49] Thompson J.F., Warsi Z.U.A. and Mastin C.W., *Numerical grid generation: Foundations and applications*, North-Holland, New York, 1985.

[50] Valiquette F. and Winternitz P., Discretization of partial differential equations preserving their physical symmetries, *J. Phys. A* **38** (2005), 9765–9783.

Chapter 5

Invariant discretization schemes using evolution–projection techniques

Alexander Bihlo[†‡] and Jean-Christophe Nave[‡]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*

[‡] *Department of Mathematics and Statistics, McGill University, 805 Sherbrooke W., Montréal (QC) H3A 2K6, Canada*

E-mail*:* [†]*bihlo@crm.umontreal.ca,* [‡]*jcnave@math.mcgill.ca*

Finite difference discretization schemes preserving a subgroup of the maximal Lie invariance group of the one-dimensional linear heat equation are determined. These invariant schemes are constructed using the invariantization procedure for non-invariant schemes of the heat equation in computational coordinates. We propose a new methodology for handling moving discretization grids which are generally indispensable for invariant numerical schemes. The idea is to use the invariant grid equation, which determines the locations of the grid point at the next time level only for a single integration step and then to project the obtained solution to the regular grid using invariant interpolation schemes. This guarantees that the scheme is invariant and allows one to work on the simpler stationary grids. The discretization errors of the invariant schemes are established and their convergence rates are estimated. Numerical tests are carried out to shed some light on the numerical properties of invariant discretization schemes using the proposed evolution–projection strategy.

5.1 Introduction

Discretization schemes for differential equations that are not solely constructed for the sake of reducing the local discretization error as much as possible, but rather to preserve some of the intrinsic properties of these differential equations have become increasingly popular over the last decades. While preserving one of these properties, namely conservation laws, led to the design of conservative discretization schemes which are quite popular in the scientific community [4, 26, 33] (and in particular in the geosciences, e.g. [18, 32]), there are other geometric features

of differential equations that can be attempted to be preserved as well that have received less attention from the side of numerical analysis so far. One of these features are symmetries of differential equations. While there have been theoretical advancements on the methodologies of finding numerical schemes that preserve the maximal Lie invariance groups of systems of differential equations over the past 20 years or so [6, 12, 21, 52], little is known about the numerical properties of these invariant schemes. A part of the problem is that while conservation laws are always properties of the solutions of a differential equation, symmetries are by definition properties of differential equations. Therefore, it is a standing question whether a discretization scheme that preserves numerically a property of a differential equation also improves the quality of the numerical solution of that discretized differential equation.

The present paper is devoted to an investigation of this question and related problems exemplified with invariant discretization schemes for the linear heat equation. The heat equation is particularly suited for this investigation as it is a canonical example in the group analysis of differential and difference equations. Moreover, there are already several studies devoted to invariant numerical schemes for this equation [2, 8, 52]. At the same time, in none of these existing works a deeper background analysis of the numerical properties (e.g. order of approximation or stability) of the developed schemes was investigated. A first account on numerical properties of invariant numerical schemes for the heat equation was given in [12], in which a numerical comparison of invariant and non-invariant schemes for the heat equation regarding accuracy was presented.

There are several reasons why less attention has been paid so far on the numerical analysis of invariant schemes (with the exception of the works [9, 12]). One of the reasons is that the field of invariant discretization schemes is still in its early stages, with new conceptual algorithms being developed only recently [4, 6–8, 12, 13, 21, 22]. Another reason is that invariant finite difference schemes generally require the use of *adaptive moving meshes*, i.e. it is necessary to include a non-trivial mesh equation in the discretization problem. Moving meshes lead to non-uniform grid point distributions and, in the multi-dimensional case, to non-orthogonal grids. The analysis of schemes on such meshes is considerably more difficult than that for related difference schemes on fixed, uniform and orthogonal meshes. Due to this second reason, most invariant numerical schemes so far have been constructed only for (1+1)-dimensional single evolution equations, as in that case moving meshes can be handled still with limited effort. Although we will be concerned with a (1+1)-dimensional equation in the present paper too, the methods used subsequently can be employed in the multi-dimensional case without substantial modification.

The new approach we propose here is to use the invariant grid equation only for a single time step and then to interpolate the solution to the regular grid. The important observation is that this interpolation can be done in an invariant way, i.e. projecting the solution does not break the invariance of the scheme itself. At the same time, the possibility to project the solution of an invariant scheme to a regular grid is highly desirable as in the multi-dimensional case a freely evolving grid can cause severe numerical problems. Moreover, for realistic numerical models, as e.g. employed in weather and climate predictions, it is in general hard to use adaptive meshes as the discretization of the governing equations is only one part of such model. Other parts are related to the numerical data assimilation, i.e. the preparation of the initial conditions for the numerical model and this step usually involves the forecasting model itself. As the assimilation of the initial conditions cannot be done on an evolving mesh (because the data are given at fixed locations only) this at once renders invariant schemes on moving meshes impractical. Equally

important, any realistic numerical model for a nonlinear system of partial differential equations has to contain subgrid-scale models, which mimic the effects of processes taking place at those scales that the numerical model is not capable of resolving explicitly [34, 35]. The construction of subgrid-scale models for non-resolved physical processes involves in general ad-hoc arguments and these arguments rely on the particular scale on which the unresolved processes take place. As a moving mesh locally changes the resolution and thus impacts what processes are explicitly resolved by a numerical model, subgrid-scale schemes have to be designed that can operate on grids with varying resolution. For realistic processes (which are usually not self-similar), this might be difficult to achieve in practice.

All what was said above objects against invariant numerical schemes for multi-dimensional systems of differential equations using freely evolving meshes. Thus, whether mathematically feasible or not, such schemes would be of less practical interest. This is why other approaches should be sought that on the one hand allow one to retain the invariance group of a system of differential equations in the course of discretization and on the other hand yield schemes that are practical to avoid the above mentioned and related problems. The proposed invariant *evolution–projection strategy* we are going to introduce below may be considered as one such approach.

The further organization of this article is as follows. The subsequent Section 5.2 features a summary and some extensions on the various methods to construct invariant discretization schemes. In Section 5.3 the heat equation along with its maximal Lie invariance group G is presented. It is discussed which subgroup G^1 of G we aim to present when constructing invariant numerical discretization schemes. The selection of G^1 is based on preserving the class of periodic boundary value problems we are focussing on. Section 5.4 contains the construction of an equivariant moving frame for G^1 along with a presentation of some lower order differential invariants of G^1. In Section 5.5 invariant discretization schemes for the heat equation in computational coordinates are found. The local discretization errors of these schemes are established in Section 5.6. In Section 5.7 we introduce the new idea of invariant interpolation schemes that can be used to project the numerical solution obtained from an invariant scheme on a moving mesh to the regular grid. The numerical analysis as well as some numerical tests for the schemes proposed in this paper are found in Section 5.8. The summary of this article is presented in the final Section 5.9.

5.2 Construction of invariant discretization schemes

The construction of invariant discretization schemes for differential equations can be seen as a part of the ongoing effort to turn group analysis into an efficient tool for the analysis of difference equations, see e.g. the review article [13]. As of now, there are three main methods that are used to construct invariant discretization schemes.

5.2.1 Difference invariant method

The first method was developed by V. Dorodnitsyn, see [2, 6, 8, 13, 52]. It uses the infinitesimal generators of one-parameter symmetry groups of the system of differential equations under consideration that span the maximal Lie invariance algebra $\mathfrak{g}$ of this system. These generators are of the form

$$\mathrm{v} = \zeta^j(x,u)\partial_{x^j} + \eta^\alpha(x,u)\partial_{u^\alpha} = \zeta(x,u)\partial_x + \eta(x,u)\partial_u,$$

where $x = (x^1, \dots, x^p)$ and $u = (u^1, \dots, u^q)$ are the tuples of independent and dependent variables, respectively. Here and in the following, the summation convention over repeated indices is used. Rather than prolonging v to higher order derivatives of u with respect to x, which is standard in the symmetry analysis of differential equations [4, 30, 42], in this method the vector fields are prolonged to all the points of the *discretization stencil*, i.e. the collection of grid points which are necessary to approximate a given system of differential equation up to a desired order. This prolongation is of the form

$$\operatorname{pr} \mathrm{v} = \sum_{i=1}^{m} \zeta(x_i, u_i)\partial_{x_i} + \eta(x_i, u_i)\partial_{u_i},$$

where $x_i = (x_i^1, \dots, x_i^p)$ and $u_i = (u_i^1, \dots, u_i^q)$, i.e. it is done by evaluating the vector field v at all m stencil points $z_i = (x_i, u_i)$ and summing up the result. An example for such a prolongation is given in Remark 5.1 in Section 5.5.

As a next step, the invariants of the group action are found by invoking the infinitesimal invariance criterion [6, 42], which in the present case is $\operatorname{pr} \mathrm{v}(I) = 0$. The functions I that fulfill this condition for all $\mathrm{v} \in \mathfrak{g}$ are termed *difference invariants*.

Once the difference invariants on the stencil space are found, one then has to assemble these invariants together to a finite difference approximation of the given system of differential equations. By construction, this procedure guarantees that the resulting numerical scheme is invariant under the symmetry group of the original system of differential equations.

The main drawback of this method is that it might be hard to find a combination of difference invariants that approximates a system of differential equations in the multi-dimensional case. The problem is, as discussed in the introduction, that invariant schemes generally require the use of moving and/or non-orthogonal grids. Formulating consistent discretization schemes using difference invariants as building blocks on moving meshes is rather challenging in higher dimensions and thus limited the application of this method to the case of single $(1+1)$-dimensional evolution equations. We stress though that this problem only enters at the stage of combining difference invariants to a discretization scheme. Computing difference invariants in the multi-dimensional case can be done as effectively as computing differential invariants for multi-dimensional problems using infinitesimal techniques.

5.2.2 Invariant moving mesh method

Retaining the invariance of finite difference schemes under the maximal Lie invariance groups of physically relevant time-dependent differential equations often requires the use of moving meshes. This is true both for the finite difference method discussed in the previous Section 5.2.1 and the moving frame method to be discussed in the next Section 5.2.3. This kind of mesh adaptation in which the number of grid points remains constant throughout the integration is referred to as r-adaptivity in the field of adaptive numerical schemes [8, 11].

The standard strategy to handle r-adaptive meshes is to regard the grid adaptation as a time-dependent mapping from a fixed reference space of *computational coordinates* to the physical space of the independent variables of the differential equation, i.e. $x = x(\xi)$ for $\xi = (\xi^1, \dots, \xi^p)$ being the computational variables. Without loss of generality, we assume that $\xi^1 = \tau = t$ is the time variable. The dependent variables u are expressed in the computational space by setting $\bar{u}(\xi) = u(x(\xi))$. For the sake of simplicity we will omit the bars henceforth.

The significance of the computational coordinates is to provide a reference frame that remains stationary and orthogonal even in the presence of grid adaptation in the physical space of coordinates. In the course of discretization the variable ξ labels the position of the grid points in the mesh and this labeling stays unchanged during the mesh adaptation. Thus, the computational variables can be interpreted physically as Lagrangian coordinates and their invariance under the motion of the grid is equivalent to the identity of fluid particles in ideal hydrodynamics.

Because by construction the grid remains orthogonal in the ξ-coordinates, the usual finite difference approximations for derivatives can be used in the space of computational variables. This simplifies both the practical implementation of the discretization method and the numerical analysis of the resulting schemes.

The expression of the initial physical system of differential equations in terms of computational variables leads to a system of equations that explicitly includes the mesh velocity x_τ, which is yet to be determined in order to close the resulting numerical scheme. A prominent strategy for determining the location of the grid points at the subsequent time level in the one-dimensional case uses the *equidistribution principle*, which in its differential form is $(\rho x_\xi)_\xi = 0$, where ρ is a monitor function that determines the areas of grid convergence and divergence. For higher-dimensional problems, equidistribution has to be combined with heuristic arguments, see [11] for more details.

The invariance of the initial differential equations is brought into the scheme by adequately specifying the monitor function ρ. In [7] it was pointed out that using monitor functions that preserving the scale-invariance of a differential equation is particularly relevant in cases where the equation is capable of developing a blow-up solution in finite time, see also [8, 10, 20]. This finding is generalized upon requiring that the monitor function is chosen in a manner such that the equidistribution principle is invariant under the same symmetry group as the original differential equation. For a number of symmetry groups this appears to be possible, see [4] for an example.

The invariant moving mesh method was recently extended in [4]. The idea of this extension is to transform the initial system of differential equations to the space of computational coordinates and to determine the form of the symmetry transformations in the computational space. The equations in the computational space are then discretized such that the resulting scheme mimics the transformation behavior of the continuous case. The main advantage of this approach is that it allows one to retain an initial conserved form of the system of differential equations and thus to numerically preserve certain conservation laws in the invariant scheme. This is relevant as preserving conservation laws in the course of invariant numerical modeling is yet a pristine problem. An exception to this is the discretization of equations that follow from variational principles, which, if done in a proper way, can lead to the simultaneous preservation of both symmetries and associated conservation laws, owing to the discrete Noether theorem. See, e.g. [8] for an example of such an invariant Lagrangian discretization.

Another advantage of the extension proposed in [4] is that it allows one to find invariant numerical schemes without the detour of difference invariants. This is essential as it can happen that the single equations in a system of differential equations cannot be approximated directly in terms on differential invariants but only in combination with other equations of that system. If this is the case it is not natural to attempt to discretize the system using difference invariants as this would lead to rather cumbersome discretization schemes.

5.2.3 Moving frame method

The third method is the most recent one [9, 10, 12, 21, 22, 31]. It relies on the notion of *equivariant moving frames* and their important property to provide a mapping that allows one to associate an invariant function to any given function. As we will mostly work with this method in the present paper, we describe it in greater detail here. We collect some important notions on moving frames below, a more comprehensive exposition can be found in the original references [9, 9, 10, 17, 18, 21].

Definition 5.1. Let G be a finite-dimensional Lie group acting on a manifold M. A *(right) moving frame* ρ is a smooth map $\rho\colon M \to G$ satisfying the equivariance property

$$\rho(g \cdot z) = \rho(z)g^{-1}, \tag{5.1}$$

for all $z \in M$ and $g \in G$.

Theorem 5.1. *A moving frame exists in the neighborhood of a point $z \in M$ if and only if the group G acts* freely *and* regularly *near z.*

Local freeness of a group action means that $\tilde{z} = g \cdot z = z$ for all z from a sufficiently small neighborhood of each point on M only holds for g being the identity transformation, which implies that all the group orbits have the same dimension. Here and throughout the paper, a tilde over a variable denotes the corresponding transformed form of that variable. *Regularity* of a group action requires that there exists a neighborhood for each point $z \in M$, which is intersected by the orbits of G into a pathwise connected subset.

When a group G does not act freely on M, its action can be made free upon extending it to a suitably high-order *jet space* $J^n = J^n(M, p)$ of M, $0 \leq n \leq \infty$. Locally, the nth order jet space of a p-dimensional submanifold of M has coordinates $z^{(n)} = (x, u^{(n)})$, where as in the previous subsections $x = (x^1, \dots, x^p)$ are considered as the independent variables, $u = (u^1, \dots, u^q)$, $q = \dim M - p$, are the dependent variables and $u^{(n)}$ collects all the derivatives of u with respect to x of order not greater than n including u as the zeroth order derivatives. In practice, the prolongation of the group action of G on J^n is implemented using the chain rule.

Moving frames are determined using a *normalization* procedure. The steps to find a moving frame for a group action G are the following: (i) Define a *cross-section* to the group orbits. A cross-section C is any submanifold $C \subset M$ of complementary dimension to the dimension r of the group orbits, i.e. $\dim C = \dim M - r$, that intersects each group orbit once and transversally. Usually coordinate cross-sections are chosen in which some of the coordinates of M (or of J^n if the group action is not free on M) are set to constants, i.e. $z^1 = c^1, \dots, z^r = c^r$. (ii) The algebraic system $\tilde{z}^1 = (g \cdot z)^1 = c^1, \dots, \tilde{z}^r = (g \cdot z)^r = c^r$ is solved for the group parameters $g = (g_1, \dots, g_r)$. The resulting expression $g = \rho(z)$ is the moving frame.

Moving frames can be used to map any given function to an invariant function by a procedure called *invariantization*.

Definition 5.2. The *invariantization* of a real-valued function $f\colon M \to \mathbb{R}$ using the (right) moving frame ρ is the function $\iota(f)$, which is defined as $\iota(f)(z) = f(g \cdot z)|_{g=\rho(z)} = f(\rho(z) \cdot z)$.

That the function $\iota(f)$ constructed in this way is indeed invariant follows from the equivariance property (5.1) of the moving frame ρ,

$$\iota(f)(g \cdot z) = f(\rho(g \cdot z)g \cdot z) = f(\rho(z)g^{-1}g \cdot z) = f(\rho(z) \cdot z) = \iota(f)(z),$$

which boils down to the definition of an invariant function I, i.e. $I(g \cdot z) = I(z)$. In practice, a function $f(z)$ is invariantized by first transforming its argument using the transformations from G and then substituting the moving frame for the group parameters. By definition, an invariant that is defined on the jet space J^n is a *differential invariant*.

Moving frames can also be constructed on a discrete space. In a finite difference approximation, derivatives of functions are approximated using a finite set of values of these functions, and all the points needed to approximate the derivatives arising in a system of differential equations are the points of the stencil introduced in Section 5.2.1. Because most of the interesting symmetries of differential equations that are broken in standard numerical schemes require the use of non-orthogonal discretization meshes, it is beneficial to both regard x and u as the dependent variables and the computational variables ξ as the independent variables as was discussed in the previous Section 5.2.2.

Sampling the tuples from the *extended computational space* $M_\xi = \{(\xi, z)\}$ at discrete points, i.e. at $(\xi_i, z(\xi_i)) = (\xi_i, z_i)$, one can introduce the space

$$M_\xi^{\diamond n} = \{(w_1, \dots, w_n) \,|\, \xi_i \neq \xi_j \text{ for all } i \neq j\},$$

where $w_i = (\xi_i, z_i)$, which can be identified as the joint product space of stencil variables. Because the identifier ξ_i of the point w_i is required to be unique, each element of $M_\xi^{\diamond n}$ only includes distinct grid points in the physical space of equation variables. The dimension of the space $M_\xi^{\diamond n}$ depends on the number of independent and dependent variables in the system of differential equations and the desired order of accuracy of the approximated derivatives.

It is possible to carry out the construction of the moving frame on $M_\xi^{\diamond n}$, i.e. to define the moving frame by an equivariant mapping $\rho_\xi^{\diamond n} \colon M_\xi^{\diamond n} \to G$, where G acts on $M_\xi^{\diamond n}$ by the product action, $g \cdot (w_1, \dots, w_n) = (g \cdot w_1, \dots, g \cdot w_n)$. Note that the extension of the group action to the computational variables ξ is trivial, i.e. they remain unaffected by G, $\tilde{\xi} = g \cdot \xi = \xi$, see [4]. The compatibility between the moving frame $\rho_\xi^{\diamond n}$ and the moving frame ρ on the space M (or an appropriate jet space J^n), i.e. that $\rho_\xi^{\diamond n} \to \rho$ in the continuous limit is assured provided that the cross-section defining the moving frame $\rho_\xi^{\diamond n}$ in the continuous limit converges to the cross-section defining the moving frame ρ. Once the moving frame is constructed on the discrete space $M_\xi^{\diamond n}$ of stencil variables, it can be used to invariantize any numerical scheme expressed in computational coordinates. This will be explicitly shown in Sections 5.5 and 5.6 where we will construct invariant schemes for the heat equation.

It is essential that the construction of the moving frame on the grid point space is carried out in terms of computational coordinates rather than physical coordinates. This can be illustrated by the following simple example.

Example 7. The Laplace equation $u_{xx}+u_{yy} = 0$ is, inter alia, invariant under the one-parameter group of rotations SO(2), $\tilde{x} = x\cos\varepsilon - y\sin\varepsilon$, $\tilde{y} = x\sin\varepsilon + y\cos\varepsilon$. Let us obtain the moving frame ρ for this group action from the normalization condition $u_x = 0$, i.e. we determine the moving frame on the first jet space $J^1(M, 2)$, $\rho = \rho(x, y, u, u_x, u_y)$. Prolonging the transformations from SO(2) to the derivative u_x leads to $\tilde{u}_{\tilde{x}} = u_x \cos\varepsilon - u_y \sin\varepsilon$ and thus the moving frame is $\varepsilon = \arctan(u_x/u_y)$.

Let us now find the product frame from the discrete normalization condition $u_x^{\mathrm{d}} = 0$. Computing u_x^{d} in the naïve way, $u_x^{\mathrm{d}} = (u_{i+1} - u_{i-1})/(x_{i+1} - x_{i-1})$, we fail as

$$\tilde{u}_{\tilde{x}}^{\mathrm{d}} = \frac{u_{i+1} - u_{i-1}}{(x_{i+1} - x_{i-1})\cos\varepsilon - (y_{i+1} - y_{i-1})\sin\varepsilon} = 0,$$

cannot be solved for the group parameter ε. On the other hand, setting $u = u(x(\xi^1, \xi^2), y(\xi^1, \xi^2))$ and expressing u_x^{d} in terms of the computational variables ξ^1, ξ^2, the normalization $u_x^{\mathrm{d}} = 0$ reads

$$\tilde{u}_{\tilde{x}}^{\mathrm{d}} = \frac{\tilde{u}_{\xi^1}^{\mathrm{d}} \tilde{y}_{\xi^2}^{\mathrm{d}} - \tilde{u}_{\xi^2}^{\mathrm{d}} \tilde{y}_{\xi^1}^{\mathrm{d}}}{\tilde{x}_{\xi^1}^{\mathrm{d}} \tilde{y}_{\xi^2}^{\mathrm{d}} - \tilde{x}_{\xi^2}^{\mathrm{d}} \tilde{y}_{\xi^1}^{\mathrm{d}}} = \frac{u_{\xi^1}^{\mathrm{d}} (x_{\xi^2}^{\mathrm{d}} \sin\varepsilon + y_{\xi^2}^{\mathrm{d}} \cos\varepsilon) - u_{\xi^2}^{\mathrm{d}} (x_{\xi^1}^{\mathrm{d}} \sin\varepsilon + y_{\xi^1}^{\mathrm{d}} \cos\varepsilon)}{x_{\xi^1}^{\mathrm{d}} y_{\xi^2}^{\mathrm{d}} - x_{\xi^2}^{\mathrm{d}} y_{\xi^1}^{\mathrm{d}}} = 0.$$

This expression can be solved for ε and it gives

$$\varepsilon = \arctan\left(\frac{u_{\xi^1}^{\mathrm{d}} y_{\xi^2}^{\mathrm{d}} - u_{\xi^2}^{\mathrm{d}} x_{\xi^1}^{\mathrm{d}}}{u_{\xi^2}^{\mathrm{d}} y_{\xi^1}^{\mathrm{d}} - u_{\xi^1}^{\mathrm{d}} x_{\xi^2}^{\mathrm{d}}}\right) = \arctan\left(\frac{u_x^{\mathrm{d}}}{u_y^{\mathrm{d}}}\right),$$

which in the continuous limit goes to $\varepsilon = \arctan(u_x/u_y)$ as required.

5.3 Lie symmetries of the heat equation

The one-dimensional linear heat transport equation is

$$u_t - u_{xx} = 0, \tag{5.2}$$

where we scaled the thermal diffusivity ν to 1, i.e. Eq. (5.2) is in non-dimensional form.

The heat equation (5.2) admits the following infinitesimal generators of one-parameter groups, which generate the maximal Lie invariance algebra $\mathfrak{g}$ of Eq. (5.2):

$$\begin{aligned} &\partial_t, \quad \partial_x, \quad u\partial_u, \quad 2t\partial_t + x\partial_x, \quad 2t\partial_x - xu\partial_u, \\ &4t^2\partial_t + 4tx\partial_x - (x^2 + 2t)u\partial_u, \quad \alpha(t,x)\partial_u, \end{aligned} \tag{5.3}$$

where α runs through the set of solutions of Eq. (5.2), see e.g. [42]. These vector fields generate (i) time-translations, (ii) space translations, (iii) scalings in u, (iv) simultaneous scalings in t and x, (v) Galilean boosts, (vi) inversions and (vii) the superposition principle symmetry.

In this paper, we will construct invariant numerical schemes for a class of initial value problems of the heat equation using periodic boundary conditions. This class of initial-boundary value problems only admits a subgroup of the symmetry group of the heat equation as inversions are no longer admitted; inversions do not send an initial value problem from the considered class to another initial value problem. The symmetries associated with the first five vector fields in (5.3) are compatible with the class of initial-boundary value problems we are interested in, i.e. they map the class of initial-boundary value problems for the heat equation under consideration to itself. This re-interpretation of symmetries of differential equations without initial and boundary conditions as equivalence transformations for a class of initial-boundary value problems was recently pointed out in [4].

In what follows we will thus focus our attention on constructing numerical schemes that preserve the symmetries generated by the first five operators of (5.3). The associated subalgebra of $\mathfrak{g}$ will be denoted by $\mathfrak{g}^1$. We do not require to preserve the linearity operator here by construction. At the same time, as will be shown in Section (5.8) the numerical schemes we propose in this paper preserve the linearity property up to the discretization error expected.

5.4 Moving frame and differential invariants for the heat equation

We determine the moving frame for the subgroup G^1 of transformations associated with the subalgebra $\mathfrak{g}^1$. Transformations of G^1 are of the form

$$\tilde{t} = e^{2\varepsilon_4}(t+\varepsilon_1), \quad \tilde{x} = e^{\varepsilon_4}(x+\varepsilon_2+2\varepsilon_5 t), \quad \tilde{u} = e^{\varepsilon_3-\varepsilon_5 x-\varepsilon_5^2 t}u. \tag{5.4}$$

Because the group action of G^1 is not free on $M=\{(t,x,u)\}$ we construct the moving frame on a suitably high-order jet space. In the present case, the group action of G^1 becomes free on $J^1=J^1(M,2)$. Thus, it is necessary to extend the transformations (5.4) to derivatives of u with respect to t and x.

Using the chain rule we can compute the transformed operators of total differentiation, which read as

$$\mathrm{D}_{\tilde{t}} = e^{-2\varepsilon_4}(\mathrm{D}_t - 2\varepsilon_5\mathrm{D}_x), \quad \mathrm{D}_{\tilde{x}} = e^{-\varepsilon_4}\mathrm{D}_x,$$

where $\mathrm{D}_x = \partial_x + u_x\partial_u + u_{tx}\partial_{u_t} + u_{xx}\partial_{u_x} + \dots$ and $\mathrm{D}_t = \partial_t + u_t\partial_u + u_{tt}\partial_{u_t} + u_{tx}\partial_{u_x} + \dots$ denote the usual operators of total differentiation. With the transformed total differentiation operators at hand it is possible to compute the transformed partial derivatives of u with respect to t and x. The transformation rules for the lowest order derivatives are

$$\begin{aligned}
&\tilde{u}_{\tilde{t}} = e^{-2\varepsilon_4+\varepsilon_3-\varepsilon_5 x-\varepsilon_5^2 t}(u_t - 2\varepsilon_5 u_x + \varepsilon_5^2 u), \quad \tilde{u}_{\tilde{x}} = e^{-\varepsilon_4+\varepsilon_3-\varepsilon_5 x-\varepsilon_5^2 t}(u_x-\varepsilon_5 u),\\
&\tilde{u}_{\tilde{x}\tilde{x}} = e^{-2\varepsilon_4+\varepsilon_3-\varepsilon_5 x-\varepsilon_5^2 t}(u_{xx} - 2\varepsilon_5 u_x + \varepsilon_5^2 u).
\end{aligned}$$

In fact, for the construction of the moving frame already the knowledge of the first order derivatives is sufficient.

We compute the moving frame for the five-parameter group of transformations of the form (5.4) using the following normalization conditions which determine a valid cross-section to the group orbits of the prolonged action of G^1 on J^1,

$$t=0, \quad x=0, \quad u=1, \quad u_t=1, \quad u_x=0. \tag{5.5}$$

The moving frame is computed by taking the transformed form of the normalization conditions, $\tilde{t}=0$, $\tilde{x}=0$, $\tilde{u}=1$, $\tilde{u}_{\tilde{t}}=1$ and $\tilde{u}_{\tilde{x}}=0$ and by solving the resulting algebraic system for the group parameters $\varepsilon_1,\dots,\varepsilon_5$. The result of this computation is the following moving frame $g=\rho(z^{(1)})$,

$$\begin{aligned}
&\varepsilon_1 = -t, \quad \varepsilon_2 = -\left(x+2t\frac{u_x}{u}\right), \quad \varepsilon_3 = -\left(\ln u - x\frac{u_x}{u} - t\frac{u_x^2}{u^2}\right),\\
&\varepsilon_4 = \ln\sqrt{\frac{u_t}{u}-\frac{u_x^2}{u^2}}, \quad \varepsilon_5 = \frac{u_x}{u}.
\end{aligned} \tag{5.6}$$

With the moving frame at hand, we can invariantize any of the partial derivatives of u with respect to t and x and thus obtain a complete set of differential invariants for the subgroup G^1 of the maximal Lie invariance group of the heat equation. As an example, invariantizing the derivative u_{xx}, i.e. computing $\iota(u_{xx})$ as $(g\cdot u_{xx})|_{g=\rho(z^{(1)})}$ we produce the differential invariant

$$\iota(u_{xx}) = \frac{uu_{xx}^2 - u_x^2}{uu_t - u_x^2}.$$

Invariantizing the heat equation, i.e. computing $\iota(u_t - u_{xx}) = 0$ and recalling that $\iota(u_t) = 1$, we obtain

$$\frac{u(u_t - u_{xx})}{uu_t - u_x^2} = 0,$$

which yields the original heat equation expressed in terms of differential invariants. This re-expression of a differential equation using the differential invariants of its symmetry group is known as the *replacement theorem* [9].

5.5 Invariant discretization of the heat equation

The invariant discretization of Eq. (5.2) cannot be done on a fixed, uniform grid. To see this, let us check the transformation behavior of the grid equation $x_i^{n+1} - x_i^n = 0$, which is the definition of a stationary grid, under the transformations (5.4). This yields

$$\tilde{x}_i^{n+1} - \tilde{x}_i^n = e^{\varepsilon_4}(x_i^{n+1} - x_i^n + 2\varepsilon_5(t^{n+1} - t^n)),$$

which is only zero in the case when $\varepsilon_5 = 0$. Stated in another way, a discretization on a fixed grid can at most preserve the symmetry subgroup of G, which is generated by the first four elements of the maximal Lie invariance algebra $\mathfrak{g}$ of the heat equation (5.3).

Thus, the discretization of (5.2) preserving G^1 will require the use of moving grids. For this reason it is convenient to express (5.2) in terms of computational coordinates initially, i.e. we set $u(\tau, \xi) = u(\tau, x(\tau, \xi))$, where ξ is the single spatial computational variable and $\tau = t$. The heat equation in this set of coordinates reads

$$u_\tau - x_\tau \frac{u_\xi}{x_\xi} - \frac{1}{x_\xi^2}\left(u_{\xi\xi} - \frac{x_{\xi\xi}}{x_\xi} u_\xi\right) = 0. \tag{5.7}$$

So as to find the invariant discretization of the heat equation in the form (5.7), we determine the moving frame in the space of stencil variables $M_\xi^{\diamond 4}$ using the discrete analogs of the normalization conditions (5.5) expressed in terms of computational coordinates.

For the sake of convenience we introduce the notation $h^+ = x_{i+1}^n - x_i^n$, $h^- = x_i^n - x_{i-1}^n$, $\Delta\tau = \tau^{n+1} - \tau^n$. The discretization stencil we use is depicted in Fig. 5.1.

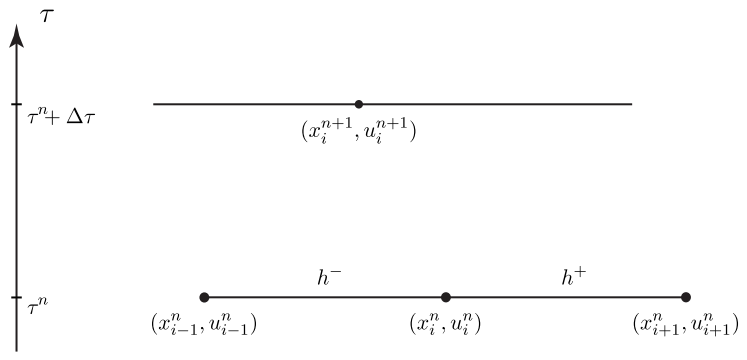

Figure 5.1: Stencil for an invariant discretization scheme of the heat equation.

The appropriate normalization conditions for a compatible moving frame $\rho_\xi^{\diamond 4}$ are

$$\tau^n = 0, \quad x_i^n = 0, \quad u_i^n = 1, \quad u_t^{\mathrm{d}} = 1, \quad u_x^{\mathrm{d}} = 0, \tag{5.8}$$

where

$$u_t^{\mathrm{d}} = \frac{u_i^{n+1} - u_i^n}{\Delta\tau} - x_\tau^{\mathrm{d}} \frac{u_{i+1}^n - u_{i-1}^n}{h^+ + h^-}, \quad u_x^{\mathrm{d}} = \frac{u_{i+1}^n - u_{i-1}^n}{h^+ + h^-}$$

are the discretizations of the first time and space derivatives expressed in computational coordinates and $x_\tau^{\mathrm{d}} = (x_i^{n+1} - x_i^n)/\Delta\tau$ is the discrete grid velocity. Replacing the single equations in the above normalization conditions by their respective transformed expressions and solving the resulting algebraic system for the group parameters we obtain the following moving frame on the space of stencil variables $M_\xi^{\diamond 4}$,

$$\begin{aligned} &\varepsilon_1 = -\tau^n, \quad \varepsilon_2 = -(x_i^n + 2\tau^n (\ln u)_x^{\mathrm{d}}), \quad \varepsilon_3 = -\left(\ln u_i^n - x_i^n (\ln u)_x^{\mathrm{d}} - \tau^n ((\ln u)_x^{\mathrm{d}})^2\right), \\ &\varepsilon_4 = \frac{1}{2} \ln \left(\frac{\exp\left(-\Delta\tau (x_\tau^{\mathrm{d}} (\ln u)_x^{\mathrm{d}} + ((\ln u)_x^{\mathrm{d}})^2)\right) u_i^{n+1} - u_i^n}{u_i^n \Delta\tau} \right), \quad \varepsilon_5 = (\ln u)_x^{\mathrm{d}}, \end{aligned} \tag{5.9}$$

where we introduced $(\ln u)_x^{\mathrm{d}} = (\ln u_{i+1}^n - \ln u_{i-1}^n)/(h^+ + h^-)$. This moving frame is compatible with the moving frame (5.6) in that it converges to (5.6) in the continuous limit $\Delta\xi \to 0$ and $\Delta\tau \to 0$ upon using

$$\begin{aligned} &h^+ = x_\xi \Delta\xi + x_{\xi\xi} (\Delta\xi)^2/2 + O(\Delta\xi^3), \quad h^- = x_\xi \Delta\xi - x_{\xi\xi} (\Delta\xi)^2/2 + O(\Delta\xi^3), \\ &x_i^{n+1} = x_i^n + x_\tau \Delta\tau + O(\Delta\tau^2), \quad u_{i+1}^n = u_i^n + u_\xi \Delta\xi + u_{\xi\xi} (\Delta\xi)^2/2 + O(\Delta\xi^3), \\ &u_{i-1}^n = u_i^n - u_\xi \Delta\xi + u_{\xi\xi} (\Delta\xi)^2/2 - O(\Delta\xi^3), \quad u_i^{n+1} = u_i^n + u_\tau \Delta\tau + O(\Delta\tau^2). \end{aligned} \tag{5.10}$$

The moving frame (5.9) can now be used to invariantize any non-invariant finite difference discretization of (5.7) on $M_\xi^{\diamond 4}$. To illustrate this, we invariantize the standard FTCS (forward in time centered in space) scheme

$$u_t^{\mathrm{d}} - \frac{4}{(h^+ + h^-)^2} \left(u_{i+1}^n + u_{i-1}^n - 2u_i^n - (h^+ - h^-) u_x^{\mathrm{d}} \right) = 0.$$

This is done by first replacing all terms by their respective transformed expressions and substituting the moving frame (5.9) for the arising group parameters. The result of this procedure is the invariant scheme

$$\begin{aligned} S = &\frac{\exp\left(-\Delta\tau \left(x_\tau^{\mathrm{d}} (\ln u)_x^{\mathrm{d}} + ((\ln u)_x^{\mathrm{d}})^2\right)\right) u_i^{n+1} - u_i^n}{\Delta\tau} - \\ &\frac{4 \left(u_{i+1}^n \left(\frac{u_{i+1}^n}{u_{i-1}^n} \right)^{-h^+/(h^+ + h^-)} + u_{i-1}^n \left(\frac{u_{i+1}^n}{u_{i-1}^n} \right)^{h^-/(h^+ + h^-)} - 2u_i^n \right)}{(h^+ + h^-)^2} = 0. \end{aligned} \tag{5.11}$$

Again, it can be checked that the above scheme (5.11) indeed converges to (5.7) in the limit of $\Delta\xi \to 0$ and $\Delta\tau \to 0$. This will be shown explicitly in Section 5.6, where we establish the order of approximation of (5.11).

So as to complete the scheme (5.11) it is necessary to determine x_i^{n+1}, which is the ingredient missing in (5.11). There are different ways to determine a grid equation, such as using the equidistribution principle as outlined in Section 5.2.2. The problem with this strategy in the present case is that while it might be beneficial from the numerical point of view, it might not be easy to obtain an invariant discretization of this principle which does not lead to a fully coupled equation–grid system. In other words, it can happen that the grid equation includes values of

u at both t^n and t^{n+1}. While this coupling is not a problem in the one-dimensional case, it can lead to a severe restriction of the applicability for multi-dimensional equations as solving the coupled equation–grid system might then be too expensive.

A G^1-invariant grid equation that circumvents the aforementioned coupling problem can be derived from the invariantization of x_i^{n+1}. This invariantization yields

$$\iota(x_i^{n+1}) = e^{\varepsilon_4}\left(x_i^{n+1} - x_i^n + \frac{2\Delta\tau}{h^+ + h^-}(\ln u_{i-1}^n - \ln u_{i+1}^n)\right),$$

where we did not explicitly substitute the frame value for ε_4. An appropriate grid is then given through $\iota(x_i^{n+1}) = 0$, or

$$M = x_i^{n+1} - x_i^n + \frac{2\Delta\tau}{h^+ + h^-}(\ln u_{i-1}^n - \ln u_{i+1}^n) = 0. \tag{5.12}$$

This grid equation is quite similar to the grid equation

$$x_i^{n+1} - x_i^n + \frac{2\Delta\tau}{h^+ + h^-}\left(\frac{h^+}{h^-}\ln\left(\frac{u_{i-1}^n}{u_i^n}\right) - \frac{h^-}{h^+}\ln\left(\frac{u_{i+1}^n}{u_i^n}\right)\right) = 0, \tag{5.13}$$

which was found in [2, 8] using the method of difference invariants. This last grid (5.13) is not only invariant under the subgroup G^1 but under the whole maximal Lie invariance group G of the heat equation. In the continuous limit, both equation (5.12) and (5.13) converge to

$$x_\tau = -\frac{2}{x_\xi}(\ln u)_\xi.$$

We have tested all our numerical schemes with both (5.12) and (5.13) and found that the resulting schemes give asymptotically the same numerical results. In fact, as in the evolution–projection strategy that will be introduced in Section 5.7 we have $h^+ = h^- = h$, Eq. (5.12) and (5.13) coincide.

Remark 5.1. While the invariantization algorithm guarantees that the scheme (5.11) is indeed invariant under the subgroup G^1 of the maximal Lie invariance group of the heat equation, the invariance can be checked in a straightforward fashion using the infinitesimal invariance criterion as invoked in the Dorodnitsyn method discussed in Section 5.2.1. Let us recall that this criterion states that an invariant I of a group action is annihilated by the associated infinitesimal generators, i.e. $\mathrm{v}(I) = 0$ for all $\mathrm{v} \in \mathfrak{g}$. Because in the present case, the invariants are defined on the stencil space with coordinates $\tau^n, \Delta\tau, x_i^n, x_{i+1}^n, x_{i-1}^n, x_i^{n+1}, u_i^n, u_{i+1}^n, u_{i-1}^n$ and u_i^{n+1}, we have to prolong the operators of $\mathfrak{g}$ accordingly. The prolongations of the first five operators of (5.3) to the variables of the stencil are

$$\partial_{\tau^n}, \quad \partial_{x_i^n} + \partial_{x_{i+1}^n} + \partial_{x_{i-1}^n} + \partial_{x_i^{n+1}}, \quad u_i^n\partial_{u_i^n} + u_{i+1}^n\partial_{u_{i+1}^n} + u_{i-1}^n\partial_{u_{i-1}^n} + u_i^{n+1}\partial_{u_i^{n+1}},$$
$$2\tau^n\partial_{\tau^n} + 2\Delta\tau\partial_{\Delta\tau} + x_i^n\partial_{x_i^n} + x_{i+1}^n\partial_{x_{i+1}^n} + x_{i-1}^n\partial_{x_{i-1}^n} + x_i^{n+1}\partial_{x_i^{n+1}}, \quad 2\tau^n(\partial_{x_i^n} + \partial_{x_{i+1}^n} +$$
$$\partial_{x_{i-1}^n}) + 2(\tau^n + \Delta\tau)\partial_{x_i^{n+1}} - x_i^n u_i^n\partial_{u_i^n} - x_{i+1}^n u_{i+1}^n\partial_{u_{i+1}^n} - x_{i-1}^n u_{i-1}^n\partial_{u_{i-1}^n} - x_i^{n+1}u_i^{n+1}\partial_{u_i^{n+1}},$$

see [6, 13] for more details. It can be checked that $\mathrm{pr}\,\mathrm{v}(S) = 0$ and $\mathrm{pr}\,\mathrm{v}(M) = 0$ hold on $S = 0$ and $M = 0$ for all the prolonged infinitesimal generators and thus $S = 0$ is a proper invariant numerical scheme and $M = 0$ an invariant grid equation.

Remark 5.2. The heat equation is a linear partial differential equation in two independent variables. One might thus consider to set up a grid equation not only for spatial but also for temporal adaptation. The reason why we refrain from spatial-temporal adaptation here is that the symmetry group G of the heat equation is compatible with flat time layers, i.e. $t^n_{i+1} - t^n_i = 0$ is a G-invariant equation. Improving the invariant numerical scheme constructed above using temporal adaptation would thus not allow one a fair comparison against the original non-invariant FTCS scheme for the heat equation. Moreover, flat time layers are well-agreed with the physics of the heat transfer problem, which affects all points of the domain simultaneously.

5.6 Numerical properties of the invariant scheme

In this section we investigate the numerical properties of the scheme (5.11) and related schemes. We start our consideration with the estimation of the local truncation error of the scheme. The study of this question is relevant because so far little is known about the relation between the order of a non-invariant scheme and its invariantized counterpart.

The discretization of the heat equation in computational coordinates (5.7) can be formally represented as

$$u^{\mathrm{d}}_\tau - x^{\mathrm{d}}_\tau \frac{u^{\mathrm{d}}_\xi}{x^{\mathrm{d}}_\xi} - \frac{1}{(x^{\mathrm{d}}_\xi)^2}\left(u^{\mathrm{d}}_{\xi\xi} - \frac{x^{\mathrm{d}}_{\xi\xi}}{x^{\mathrm{d}}_\xi} u^{\mathrm{d}}_\xi\right) = 0, \tag{5.14}$$

where in the present case we assume that derivatives are approximated with the aid of a standard FTCS scheme. More general schemes will be considered after the order of the invariantized FTCS scheme is established.

Theorem 5.2. *The order of the invariant scheme* (5.11) *is the same as the order of the scheme* (5.14)*, namely first order in time and second order in space, provided that an Euler forward step and second order centered differences are used to approximate the time and space derivatives arising in both the differential equation* (5.14) *and the normalization conditions* (5.8)*.*

Proof. Invariantizing the scheme (5.14) using the normalization conditions (5.8) leads to

$$1 - \frac{4}{\iota((x^{\mathrm{d}}_\xi)^2)}\iota(u^{\mathrm{d}}_{\xi\xi}) = 0, \tag{5.15}$$

where $\iota(f)(z)$ denotes the invariantization of the function $f(z)$. By definition, invariantization of a function $f(z)$ means to transform the argument z and plug in the moving frame for the group parameters. In the present case, the transformed form of (5.15) can be written as

$$1 - \frac{4e^{\varepsilon_3 - \varepsilon_5 x - \varepsilon_5^2 t - 2\varepsilon_4}}{(x^{\mathrm{d}}_\xi)^2}(e^{-\varepsilon_5 h^+}u^n_{i+1} + e^{\varepsilon_5 h^-}u^n_{i-1} - 2u^n_i) = 0.$$

Using the normalization condition $u^n_i = 1$ we obtain that

$$\tilde{u}^n_i = 1 = e^{\varepsilon_3 - \varepsilon_5 x - \varepsilon_5^2 t}u^n_i$$

and thus the last expression can be recast as

$$e^{2\varepsilon_4}u^n_i - \frac{4}{(h^+ + h^-)^2}(e^{-\varepsilon_5 h^+}u^n_{i+1} + e^{\varepsilon_5 h^-}u^n_{i-1} - 2u^n_i) = 0. \tag{5.16}$$

Let us now determine the local discretization error in the parameter ε_5. The respective moving frame component is

$$\varepsilon_5 = \frac{\ln u^n_{i+1} - \ln u^n_{i-1}}{h^+ + h^-},$$

which upon using (5.10) expands to

$$\varepsilon_5 = \frac{1}{x_\xi}\frac{u_\xi}{u^n_i} + O(\Delta\xi^2). \tag{5.17}$$

Substituting ε_5 into the second term of Eq. (5.16) and expanding the exponential functions in the same term into Taylor series, we obtain after some rearranging

$$e^{2\varepsilon_4}u^n_i - \frac{1}{x^2_\xi\Delta\xi^2 + O(\Delta\xi^4)}\Bigg(u^n_{i+1} + u^n_{i-1} - 2u^n_i - \varepsilon_5\Bigg(x_\xi\Delta\xi(u^n_{i+1} - u^n_{i-1}) + \frac{1}{2}x_{\xi\xi}\Delta\xi^2(u^n_{i+1} + u^n_{i-1})\Bigg) + \frac{1}{2}\varepsilon^2_2 x^2_\xi\Delta\xi^2(u^n_{i+1} + u^n_{i-1}) + O(\Delta\xi^4)\Bigg) = 0.$$

This can be further simplified to

$$e^{2\varepsilon_4}u^n_i - \frac{1}{x^2_\xi}\left(u_{\xi\xi} - \frac{u^2_\xi}{u^n_i} - \frac{x_{\xi\xi}}{x_\xi}u_\xi\right) + O(\Delta\xi^2) = 0. \tag{5.18}$$

It now remains to expand the first term in Eq. (5.18). The moving frame component for ε_4 in (5.9) can be recast as

$$e^{2\varepsilon_4}u^n_i = \frac{\exp\left(-\Delta\tau\left(\dfrac{x^{n+1}_i - x^n_i}{\Delta\tau}\dfrac{\ln u^n_{i+1} - \ln u^n_{i-1}}{h^+ - h^-} + \left(\dfrac{\ln u^n_{i+1} - \ln u^n_{i-1}}{h^+ - h^-}\right)^2\right)\right)u^{n+1}_i - u^n_i}{\Delta\tau}$$

Using $x^{n+1}_i = x^n_i + x_\tau\Delta\tau + O(\Delta\tau^2)$ and $u^{n+1}_i = u^n_i + u_\tau\Delta\tau + O(\Delta\tau^2)$ and again expanding the exponential function into a Taylor series, we derive

$$e^{2\varepsilon_4}u^n_i = u_\tau - x_\tau\frac{u_\xi}{x_\xi} - \frac{1}{x^2_\xi}\frac{u^2_\xi}{u^n_i} + O(\Delta\tau, \Delta\xi^2).$$

Plugging this into Eq. (5.18) we arrive at

$$u_\tau - x_\tau\frac{u_\xi}{x_\xi} - \frac{1}{x^2_\xi}\left(u_{\xi\xi} - \frac{x_{\xi\xi}}{x_\xi}u_\xi\right) + O(\Delta\tau, \Delta\xi^2) = 0,$$

which completes the proof of the theorem. □

A more general statement is the following one:

Theorem 5.3. *The order of spatial discretization of an invariant finite difference scheme for the heat equation in computational variables equals the order $p \in \mathbb{N}$ of the spatial discretization of the associated non-invariant finite difference scheme provided that centered differences of order p are used to approximate both the derivatives in the heat equation and in the normalization conditions* (5.8).

Proof. In view of the general form (5.15) of the invariantization of scheme (5.14), we study the invariantization of the terms x_ξ and $u_{\xi\xi}$.

The invariantization of $(x^{\mathrm{d}}_\xi)^2 = (x_\xi + O(\Delta\xi^p))^2$ is $\iota((x^{\mathrm{d}}_\xi)^2) = e^{2\varepsilon_4}(x^2_\xi + O(\Delta\xi^p))$ and is of the same order p if, as required, the moving frame component ε_4 stems from the approximation of $u^{\mathrm{d}}_\tau = 1$ using pth order accuracy and thus only includes approximations of derivatives with that accuracy.

Let us now investigate the invariantization of discretizations of $u_{\xi\xi}$. The general form of a centered difference approximation of $u_{\xi\xi}$ of even order p is

$$u_{\xi\xi} = \frac{1}{\Delta\xi^2} \sum_{j=-p/2}^{p/2} c^2_{p,j} u^n_j + O(\Delta\xi^p),$$

where $c^2_{p,j} = 2c^1_{p,j}/j$, $j \in A = \{-p/2, \dots, -1, 1, \dots, p/2\}$, $c^2_{p,0} = -2\sum_{i=1}^{p/2} 1/i^2$ and

$$c^1_{p,j} = \frac{(-1)^{j+1}(p/2)!^2}{j(p/2+j)!(p/2-j)!}, \quad j \in A$$

and $c^1_{p,0} = 0$ are the coefficients from the pth order approximation of u_ξ, i.e.

$$u_\xi = \frac{1}{\Delta\xi} \sum_{j=-p/2}^{p/2} c^1_{p,j} u^n_j + O(\Delta\xi^p).$$

See [17] for a discussion of the algorithm for finding the weights $c^k_{p,j}$ in higher-order finite difference approximations of the kth derivative of u. The invariantization of $u_{\xi\xi}$ is

$$\iota(u_{\xi\xi}) = \frac{1}{\Delta\xi^2} \sum_{j=-p/2}^{p/2} c^2_{p,j} \exp(\varepsilon_3 - \varepsilon_5 x^n_j - \varepsilon^2_5 \tau^n) u^n_j,$$

or, upon using the normalization condition $u^n_i = 1$,

$$\iota(u_{\xi\xi}) = \frac{1}{\Delta\xi^2} \frac{1}{u^n_i} \sum_{j=-p/2}^{p/2} c^2_{p,j} \exp(-\varepsilon_5 \Delta x_j) u^n_j, \tag{5.19}$$

where we expand

$$\Delta x_j = x^n_j - x^n_i = \sum_{k=1}^{\infty} \frac{(j\Delta\xi)^k}{k!} \frac{\partial^k x}{\partial \xi^k}, \quad u^n_i = \sum_{l=0}^{\infty} \frac{(j\Delta\xi)^l}{l!} \frac{\partial^l u}{\partial \xi^l}.$$

Using the expressions for Δx_j and u^n_i, the expression (5.19) can be expanded and rearranged in powers of $j\Delta\xi$ in the form

$$\iota(u_{\xi\xi}) = \frac{1}{\Delta\xi^2} \frac{1}{u^n_i} \sum_{j=-p/2}^{p/2} \sum_{k=0}^{\infty} (-1)^k c^2_{p,j} A_k (j\Delta\xi)^k,$$

where

$$A_2 = \frac{1}{2} \left(u_{\xi\xi} - \varepsilon_5 (2 x_\xi u_\xi + x_{\xi\xi} u^n_i) + \varepsilon^2_5 x^2_\xi u^n_i \right).$$

The expressions for A_k, $k \neq 2$, are not required subsequently. The proof is completed upon substituting for ε_5 the corresponding moving frame component (which is of order p if the normalization $u_x^{\mathrm{d}} = 0$ is approximated with pth order accuracy) and by noting that

$$\sum_{j=-p/2}^{p/2} c_{p,j}^2 j^k = \begin{cases} 0 & \text{for } k \in \{0, 1, 3, \dots p+2, 2n\}, \quad n \in \mathbb{N} \\ 2 & \text{for } k = 2 \\ c_k \neq 0 & \text{else} \end{cases}$$

where the precise values of the constants c_k follow from evaluating the respective sums. □

The scheme (5.11) is only of first order in time $\tau = t$. To construct a scheme that is second order in time, we can start with a non-invariant scheme (5.14) and discretize the time derivative u_τ^{d} with second order accuracy, i.e. we set $u_\tau^{\mathrm{d}} = (u_i^{n+1} - u_i^{n-1})/(2\Delta\tau)$, where u_i^{n-1} is the value of u at the previous time step $\tau^{n-1} = \tau^n - \Delta\tau$. It is now necessary to check whether invariantizing this leapfrog discretization leads to an invariant scheme that is also second order in time.

Theorem 5.4. *Invariantization of the scheme* (5.14) *in which a leapfrog step and second order centered differences are used to approximate the time and space derivatives, leads to an invariant scheme that is both second order in time and space provided that the normalization conditions* (5.8) *are approximated using discretizations that are of second order.*

Proof. To prove this theorem it is sufficient to establish the order of the first term in Eq. (5.18). We proceed in an analog manner as in the proof of Theorem 5.2, i.e. we discretize the normalization condition $u_\tau^{\mathrm{d}} = 1$ but now with second order accuracy. This yields

$$\tilde{u}_{\tilde{\tau}}^{\mathrm{d}} = \frac{e^{\varepsilon_3 - \varepsilon_5 x_i^n - \varepsilon_5^2 \tau^n}}{2e^{2\varepsilon_4}\Delta\tau} \left(e^{-\varepsilon_5 x_\tau \Delta\tau - \varepsilon_5^2 \Delta\tau} u_i^{n+1} - e^{\varepsilon_5 x_\tau \Delta\tau + \varepsilon_5^2 \Delta\tau} u_i^{n-1} \right) = 1.$$

Using the normalization condition $u_i^n = 1$ as before and expanding the exponential functions we derive

$$e^{2\varepsilon_4} u_i^n = \frac{1}{2\Delta\tau} \left(u_i^{n+1} - u_i^{n-1} - \Delta\tau(\varepsilon_5 x_\tau + \varepsilon_5^2)(u_i^{n+1} + u_i^{n-1}) \right) + O(\Delta\tau^2)$$

and upon noting that $u_i^{n+1} + u_i^{n-1} = 2u_i^n + O(\Delta\tau^2)$ we obtain

$$e^{2\varepsilon_4} u_i^n = u_\tau - x_\tau \frac{u_\xi}{x_\xi} - \frac{1}{x_\xi^2} \frac{u_\xi^2}{u_i^n} + O(\Delta\tau^2, \Delta\xi^2),$$

where we have substituted the expression (5.17) for ε_5. Plugging this result into Eq. (5.18) completes the proof of the theorem. □

The actual form of the resulting invariant leapfrog scheme is

$$\frac{\exp\left(-\Delta\tau \left(\hat{x}_\tau^{\mathrm{d}} (\ln u)_x^{\mathrm{d}} + ((\ln u)_x^{\mathrm{d}})^2\right)\right) u_i^{n+1} - \exp\left(\Delta\tau \left(\check{x}_\tau^{\mathrm{d}} (\ln u)_x^{\mathrm{d}} + ((\ln u)_x^{\mathrm{d}})^2\right)\right) u_i^{n-1}}{2\Delta\tau} - \frac{4\left(u_{i+1}^n \left(\frac{u_{i+1}^n}{u_{i-1}^n} \right)^{-h^+/(h^+ + h^-)} + u_{i-1}^n \left(\frac{u_{i+1}^n}{u_{i-1}^n} \right)^{h^-/(h^+ + h^-)} - 2u_i^n \right)}{(h^+ + h^-)^2} = 0, \tag{5.20}$$

where $\hat{x}_\tau^{\mathrm{d}} = (x_i^{n+1} - x_i^n)/\Delta\tau$ and $\check{x}_\tau^{\mathrm{d}} = (x_i^n - x_i^{n-1})/\Delta\tau$.

Higher order in time schemes can be constructed upon invariantizing multi-stage schemes. Combining this result with the result established in Theorem 5.3 we have found the following:

Corollary 5.1. *Invariantizing a non-invariant finite difference scheme for the heat equation in computational coordinates preserves the spatial and temporal order of the initial non-invariant finite difference scheme provided that centered differences are used and the normalization conditions for the moving frame are discretized with the same order as the respective derivatives in the non-invariant finite difference scheme.*

5.7 Invariant interpolation schemes

A common property of invariant numerical schemes for evolution equations possessing a nontrivial maximal Lie invariance group is that it is not possible to use a fixed, orthogonal discretization mesh. The continuous evolution of the mesh, if not handled properly, can lead to several undesirable properties, such as an overly strong concentration of grid points in certain regions and therefore too poor a resolution in other parts of the integration domain. The problem gets worse in the multi-dimensional case where mesh tangling or strongly skewed meshes can occur. But even if the mesh movement can be managed in an optimal way there are various physical problems for which continuously adapting grids pose a severe challenge. An example for this are practically all models that are in operational use in weather and climate prediction. These models employ sophisticated data assimilation strategies and are coupled to subgrid-scale parameterizations that aim to mimic the effects of unresolved processes on the grid scale variables. Attempting to make use of data assimilation or parameterization schemes on moving meshes is not only a technical problem that would cause a significant computational overhead compared to standard schemes but also a conceptual challenge for it is unclear on how to design parameterization schemes that can operate on grids with varying resolution. In order to promote the ideas of invariant numerical discretization schemes beyond their application to simple evolution equations it is thus instructive to study possible ways of overcoming the limitations imposed by the requirement of using moving meshes.

One straightforward idea is to use invariant schemes on fixed (i.e. non-invariant grids). As was shown in [21] this can lead to improved numerical solutions compared to non-invariant integrators, while still being excelled by the results that can be obtained using completely invariant schemes. On the other hand, if moving (invariant) meshes are not tractable for a particular class of problems, preserving the invariance of a system of differential equations at least for the discretization of the system itself might be a possible trade-off to take.

Another idea is to use an *evolution–projection strategy*, which will be proposed in the following. This concept relies on using the invariant scheme for the system of differential equations together with the invariant mesh equations for a single time step followed by the projection of the numerical solution back to the regular mesh. A similar strategy has proven successful in semi-Lagrangian time integration schemes [33].

In the present case, the projection step can be practically realized by using *interpolation*. Obviously, any standard interpolation scheme can be used to map the numerical solution u_i^{n+1} defined at x_i^{n+1} to the uniformly spaced ξ-grid. This, however, can break the invariance of the numerical scheme as a whole and so the question arises whether it is possible to accomplish the interpolation step in a symmetry-preserving fashion.

In the following we discuss two possible ways of formulating *invariant interpolation schemes*, both of which can be used for finding interpolations that allow the re-mapping of the numerical solution on a moving mesh to a fixed, Cartesian, equally-spaced grid. These ways are the

invariantization of non-invariant interpolation schemes with moving frames and the construction of *interpolations using difference invariants.*

Invariantization of interpolation schemes. The moving frame constructed in the course of invariantizing a finite difference scheme can also be used to invariantize a certain interpolation method. We exemplify this idea by invariantizing the formula for linear interpolation,

$$u_i^{n+1}(y) = u_i^{n+1} + (y - x_i^{n+1})\frac{u_{i+1}^{n+1} - u_i^{n+1}}{x_{i+1}^{n+1} - x_i^{n+1}},$$

where $y \in [x_i^{n+1}, x_{i+1}^{n+1}]$. The invariantization of this expression using a moving frame associated with G^1 yields the invariant interpolation formula

$$u_i^{n+1}(y) = U_i^{n+1} + (y - x_i^{n+1})\frac{U_{i+1}^{n+1} - U_i^{n+1}}{x_{i+1}^{n+1} - x_i^{n+1}}, \quad U_i^{n+1} = \exp((\ln \hat{u}_{\hat{x}})^{\mathrm{d}}(y - x_i^{n+1}))u_i^{n+1}. \tag{5.21}$$

Note that we have used a slightly different moving frame for the invarianization as we have used for invariantizing the finite difference scheme for the heat equation. Specifically, this moving frame is constructed by replacing the normalization condition $u_x^{\mathrm{d}} = 0$ with $\hat{u}_{\hat{x}}^{\mathrm{d}} = 0$. The reason for this is that the moving frame used earlier yielded $\varepsilon_5 = (\ln u_x)^{\mathrm{d}}$, i.e. it involves the solutions of u_i at the time step τ^n rather than at $\tau^n + \Delta\tau$. Irrespectively of what normalization is used, both interpolations are invariant. Setting $y = \xi_i$ in the above interpolation formula yields u_i^{n+1} on the regular computational grid. Note that the interpolation (5.21) is consistent in that $u_i^{n+1}(x_i^{n+1}) = u_i^{n+1}$ and $u_i^{n+1}(x_{i+1}^{n+1}) = u_{i+1}^{n+1}$.

In a similar manner more sophisticated interpolation schemes can be invariantized. In the following, we will use the invariantization of quadratic interpolation. Usual quadratic interpolation is based on the expression

$$u^{n+1}(y) = u_{i-1}^{n+1}L_{i-1}(y) + u_i^{n+1}L_i(y) + u_{i+1}^{n+1}L_{i+1}(y), \quad L_j(y) = \prod_{\substack{k=i-1 \\ k\neq j}}^{i+1} \frac{y - x_k^{n+1}}{x_j^{n+1} - x_k^{n+1}}, \tag{5.22}$$

where $y \in [x_{i-1}^{n+1}, x_{i+1}^{n+1}]$ and $L_j(y)$ are the Lagrangian interpolation polynomials. Invariantizing this formula using the same moving frame as above we get

$$u^{n+1}(y) = U_{i-1}^{n+1}L_{i-1}(y) + U_i^{n+1}L_i(y) + U_{i+1}^{n+1}L_{i+1}(y), \tag{5.23}$$

where $U_i = \exp((\ln \hat{u}_{\hat{x}})^{\mathrm{d}}(\hat{x} - x_i^{n+1}))u_i^{n+1}$, as in the case of the invariant linear interpolation (5.21). Numerical examples using the invariant quadratic interpolation will be given in Section 5.8.

Interpolation using difference invariants. The product frame on the grid point space allows invariantizing the elementary variables x_i^n and u_i^n, which yields the system of joint invariants. In the continuous limit these invariants take the normalization values chosen for x and u to construct the usual moving frame ρ [17]. On the other hand, on the discrete space $M_\xi^{\diamond n}$ we only normalize one x_i^n (i, n fixed) among all the grid points x_l^k and the analog statement is true for the associated values u_l^k. This means that the joint invariants $\iota(x_l^k)$ and $\iota(u_l^k)$, $l \neq i, k \neq n$, are nontrivial and can be used to assemble invariant interpolation schemes.

In the present case, while we have normalized $u_i^n = 1$ in the course of constructing the moving frame $\rho^{\diamond 4}$, we are free to use the moving frame to invariantized any u_l^k where $l \neq i, k \neq n$ and this will yield a proper (nontrivial) invariant on the discrete space $M_\xi^{\diamond 4}$. As above, we again

recompute the moving frame for G^1 by replacing the normalization conditions $u_i^n = 1$ and $u_x^{\mathrm{d}} = 0$ with $u_i^{n+1} = 0$ and $\hat{u}_{\hat{x}}^{\mathrm{d}} = 0$, respectively, which yields new expressions for the moving frame components of ε_3 and ε_5 given by

$$\varepsilon_3 = -((\ln u_i^{n+1} - x_i^{n+1}(\ln \hat{u}_{\hat{x}})^{\mathrm{d}} - \tau^{n+1}((\ln u_{\hat{x}})^{\mathrm{d}}))^2), \quad \varepsilon_5 = (\ln \hat{u}_{\hat{x}})^{\mathrm{d}}.$$

Using this modified moving frame, we then invarianize the variable $u^{n+1}(\xi_i)$, which is the sought value of u at the point (τ^{n+1}, ξ_i) of the computational domain. This invariantization yields

$$\iota(u^{n+1}(\xi_i)) = \frac{u^{n+1}(\xi_i)}{u_i^{n+1}} \exp((\ln \hat{u}_{\hat{x}})^{\mathrm{d}}(x_i^{n+1} - \xi_i)).$$

Because in the continuous limit the invariantization of $u^{n+1}(\xi_i)$ must reproduce the normalization condition $u = 1$, we restrict the difference invariant to the manifold $\iota(u^{n+1}(\xi_i)) = 1$. The invariant interpolation is thus

$$u^{n+1}(\xi_i) = \exp((\ln \hat{u}_{\hat{x}})^{\mathrm{d}}(\xi_i - x_i^{n+1}))u_i^{n+1} \tag{5.24}$$

and it is again consistent as $u^{n+1}(x_i^{n+1}) = u_i^{n+1}$. More accurate interpolations could be constructed by combining the invariants $\iota(x_l^k)$ and $\iota(u_l^k)$ in a suitable way.

The advantage of the interpolation methods introduced in this section is that they are invariant under the group G^1, i.e. using these interpolation formulas to map u_i^{n+1} back to the ξ-grid does not break the invariance of the numerical schemes for the heat equation, while still allowing one to use a regular grid. Invariant interpolations thus allow avoiding the complications that moving meshes impose on the applicability of symmetry-preserving finite difference discretizations.

5.8 Numerical verification

In order to verify the accuracy predicted above for the various schemes proposed, we set-up the following problem. On a periodic domain $x \in [0, 2\pi[$, consider

$$u_t = u_{xx},$$
$$u\,(x, t = 0) = \sin(x - 1) + 2.$$

On a sequence of grids with $N \in \{2, 4, 8, \ldots, 256\}$, the number of grid points, we compute the error in the maximum norm between the numerical solution and the exact solution at $t = 1$. The time step $\Delta\tau$ is taken as proportional to h^2, $h = h^+ = h^-$, in all simulations.

In each of the following figures, we plot the reference line corresponding to $\mathcal{O}\left(h^2\right)$ dash–dotted, and the L_∞ error as black line where,

$$\|E\|_{L_\infty} = \max_{x \in [0, 2\pi]} |u\,(x, 1) - u_{\mathrm{exact}}\,(x, 1)|\,.$$

Note that for all approaches described below we expect a second order convergence since the numerical scheme being used is of second order, its invariantization was shown to preserve this order and the quadratic interpolation and its invariantization is also of second order.

5.8.1 Invariant scheme without projection

In this test run we use the scheme (5.11) without projection. As a result, the solution is evolving along the trajectories of the grid equation (5.12). Since the spacing between trajectories is not constant, i.e. h^+ and h^- are changing in time, we choose to plot the error versus $1/N$. In Fig. 5.2, we observe the second order convergence expected.

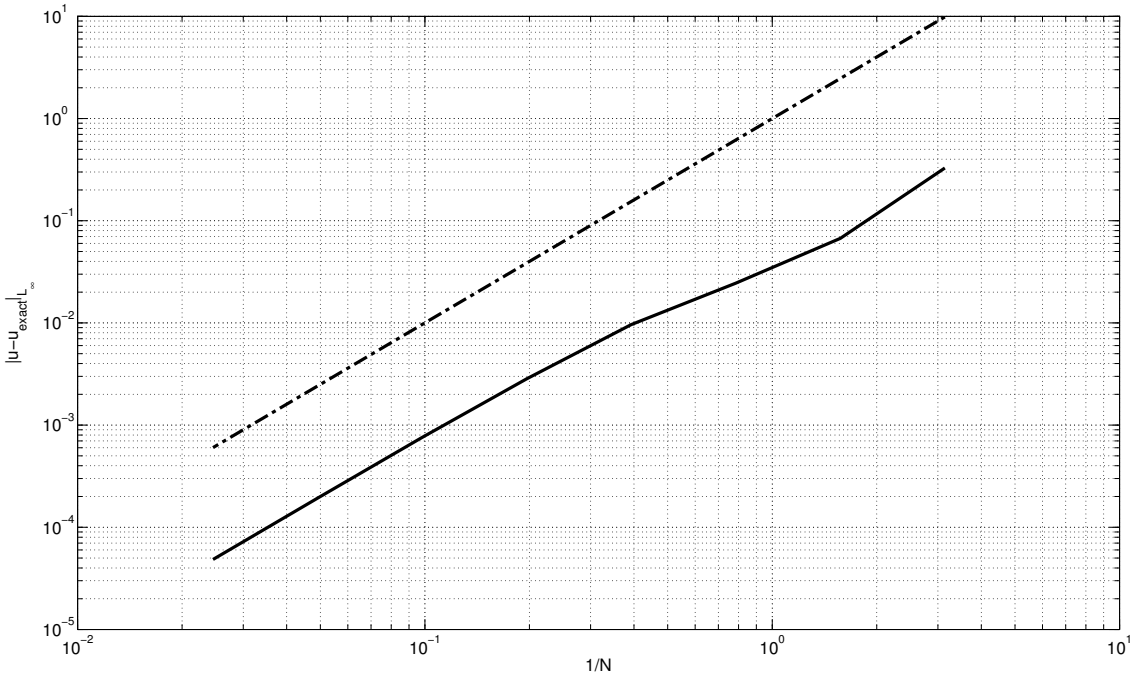

Figure 5.2: Convergence plot for the invariant scheme (5.11) with invariant grid equation (5.12) and without re-mapping.

5.8.2 Invariant scheme with non-invariant quadratic interpolation

In this scheme we interpolate the solution of the invariant scheme (5.11) at every step back onto the regular grid using standard Lagrange quadratic interpolation (5.22). As a result, we have the solution on a regular grid with step-size $h = 1/N$. In Fig. 5.3, we observe the second order convergence expected.

5.8.3 Invariant scheme with invariant quadratic interpolation

In this scheme, we interpolate the solution of the invariant scheme (5.11) at every step back onto the regular grid using the invariant Lagrange quadratic interpolation (5.23) described in the previous section. As for the case above, at each time step we have the solution on a regular grid with step-size $h = 1/N$. In Fig. 5.4, we observe the second order convergence expected.

5.8.4 Linearity preservation in the invariant numerical scheme

Linearity is not preserved by construction in the schemes proposed. For this reason it is instructive to check numerically whether or not linearity is preserved in the fully invariant scheme.

Consider the initial value problem,

$$u_t = u_{xx}$$
$$u\,(x, t = 0) = (\sin(x - 1) + 2) + (\cos x + 2),$$

the solution of which we call u_{exact}. We then solve numerically the following two equations

$$u^a_t = u^a_{xx} \text{ with } u^a(x, 0) = \sin(x - 1) + 2,$$
$$u^b_t = u^b_{xx} \text{ with } u^b(x, 0) = \cos x + 2,$$

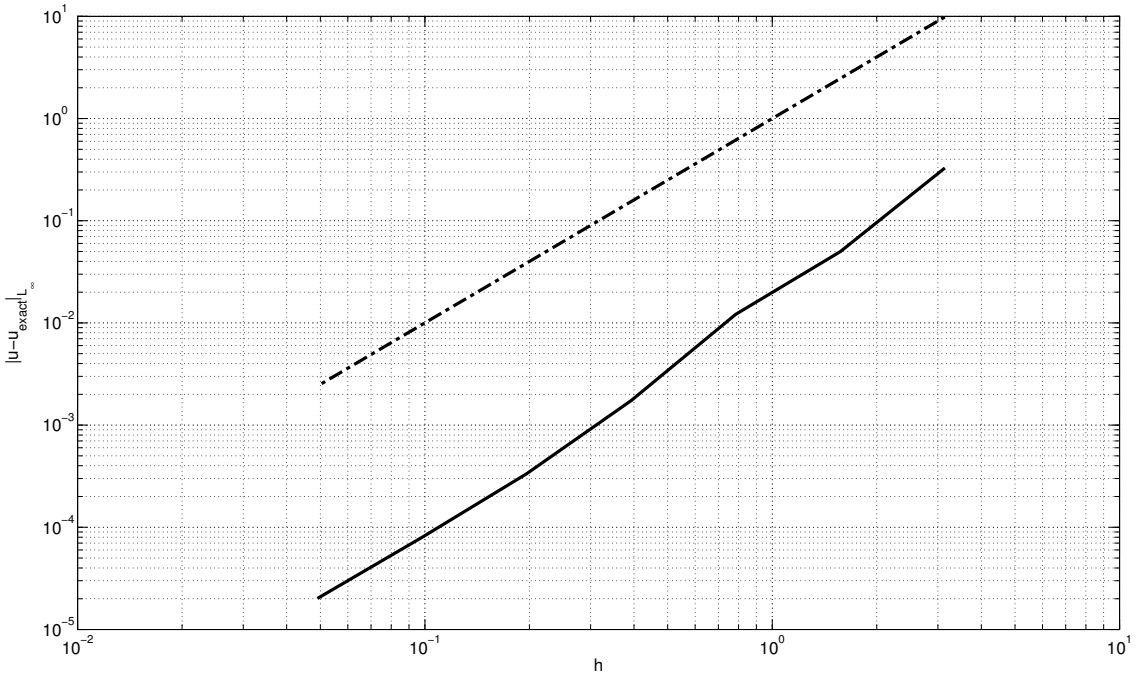

Figure 5.3: Convergence plot for the invariant scheme (5.11) with invariant grid equation (5.12) using non-invariant quadratic interpolation (5.22) as projection.

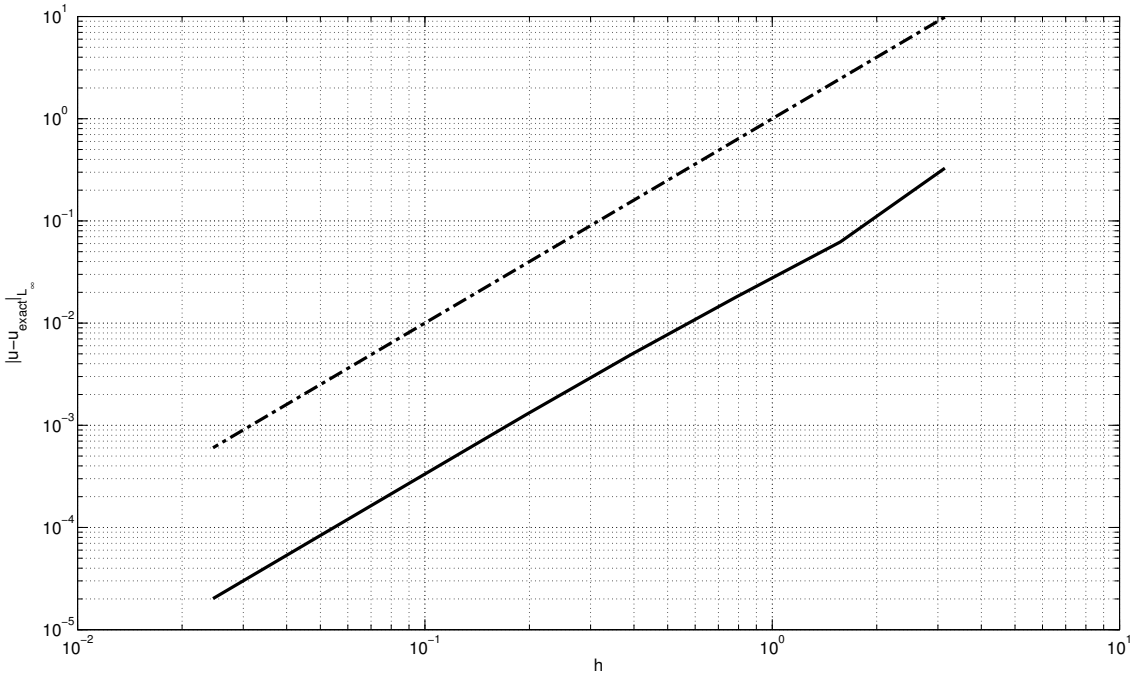

Figure 5.4: Convergence plot for the invariant scheme (5.11) with invariant grid equation (5.12) using invariant quadratic interpolation (5.23) as projection.

and define $u_s = u^a + u^b$.

Fig. 5.5 depicts the L_∞ error between u_{exact} and u_s. We observe a convergence rate of second order. In other words, despite the fully invariant numerical scheme does not explicitly preserve

the symmetry associated with the linear superposition principle, we observe that the linearity property is preserved approximately to the order of the method.

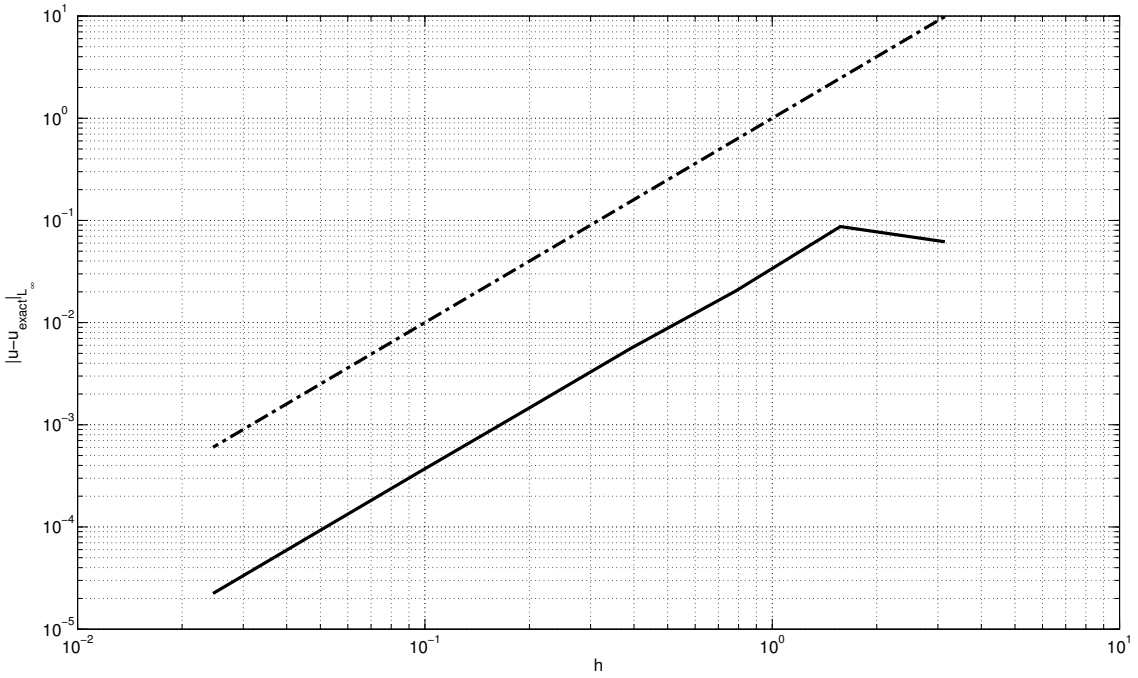

Figure 5.5: Convergence plot for the linearity test using the invariant scheme (5.11) with invariant grid equation (5.12) using invariant quadratic interpolation (5.23) as projection.

5.9 Conclusions

In this paper we construct invariant discretization schemes using the method of invariantization via equivariant moving frames. The advantage of this technique is that it allows one to start with a given non-invariant scheme and convert this initial scheme into a finite difference approximation of a system of differential equations $\mathcal{L}$ that is invariant under the same maximal Lie invariance group G (or a suitably chosen subgroup of G) as admitted by $\mathcal{L}$.

The possibility of converting non-invariant numerical schemes into invariant discretizations may lead to the overly optimistic speculation that the schemes constructed by invariantization could be easily included in existing numerical models using the original scheme. The hurdle preventing this in practice is that preserving symmetry groups of systems of evolution equations more complicated than scalings or translations requires the use of moving grids. Converting numerical models that use standard discretization schemes based on fixed lattices to (invariant) discretization schemes on moving meshes is not an easy task. At the same time, rewriting numerical models from scratch for the simulation of involved physical processes using symmetry-preserving schemes might be a time-consuming and costly task too and it not certain that this is feasible at all. Moreover, it is as of now unclear whether preserving symmetries in numerical schemes for multi-dimensional systems of partial differential equations gives enough added value compared to standard schemes that one might justify such an undertaking in practice.

This is why one relies on finding methods allowing one to efficiently include invariant discretization schemes into existing numerical models without the need to rewrite new models

from scratch that incorporate the invariance methodology. The method proposed in this article solves this problem by breaking the integration procedure into two steps, the time-stepping using the invariant numerical scheme with an invariant numerical grid equation and the projection (interpolation) of the results obtained at intermediate grid points to the regular mesh. This interpolation can be done in an invariant way by applying the moving frame map used to invariantize the initial discretization scheme also to a particular interpolation method. An alternative is to assemble the invariant interpolation method using joint invariants. Either way, it is worthwhile pointing out that interpolations requiring only data given at a single time level are already invariant under most symmetry groups as admitted by physical systems of differential equations. Thus, invariantization of interpolation formulas will often only lead to minor modifications of the initial interpolation method chosen and the influence on the numerical solution might be rather small. In the numerical tests carried out above for the heat equation, the difference in the convergence properties we found when using invariant or non-invariant interpolation methods is indeed small although using the invariant interpolation gave slightly better numerical results. This is encouraging and the reason why we plan to further investigate invariant numerical schemes using the projection procedure.

We illustrate the evolution–projection strategy by integrating the one-dimensional linear heat equation with an invariant numerical scheme. The heat equation has been studied quite extensively in light of its invariance properties and in particular it is a standard model for the construction of invariant numerical schemes [2, 8, 52]. At the same time, a comprehensive numerical analysis of such schemes was not given before and thus seems relevant to be reported. This is another aim of the present paper. Again, the analysis of numerical properties of discretization schemes is considerably easier if one can use non-evolving meshes.

Further work we intend to do is to employ the evolution–projection strategy to multi-dimensional systems of differential equations using both higher-order discretization and interpolation schemes.

Acknowledgements

The authors thank Professor Roman Popovych for valuable discussions and careful reading of the manuscript. The valuable remarks of the anonymous referees are much appreciated. This research was supported by the Austrian Science Fund (FWF), project J3182–N13 (AB). JCN wishes to acknowledge partial support from the NSERC Discovery Program, and the National Science Foundation through grant DMS-0813648.

Bibliography

[1] Bakirova M.I., Dorodnitsyn V.A. and Kozlov R.V., Symmetry-preserving discrete schemes for some heat transfer equations, *J. Phys. A* **30** (1997), 8139–8155.

[2] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[3] Bluman G. and Kumei S., *Symmetries and differential equations*, Springer, New York, 1989.

[4] Bridges T.J. and Reich S., Numerical methods for Hamiltonian PDEs, *J. Phys. A* **39** (2006), 5287–5320.

[5] Budd C. and Dorodnitsyn V.A., Symmetry-adapted moving mesh schemes for the nonlinear Schrödinger equation, *J. Phys. A* **34** (2001), 10387–10400.

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

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

[8] Budd C.J. and Iserles A., Geometric integration: numerical solution of differential equations on manifolds, *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **357** (1999), 945–956.

[9] Cheh J., Olver P.J. and Pohjanpelto J., Algorithms for differential invariants of symmetry groups of differential equations, *Found. Comput. Math.* **8** (2008), 501–532.

[10] Chhay M. and Hamdouni A., A new construction for invariant numerical schemes using moving frames, *C. R. Mecanique* **338** (2010), 97–101.

[11] Chhay M., Hoarau E., Hamdouni A. and Sagaut P., Comparison of some Lie-symmetry-based integrators, *J. Comput. Phys.* **230** (2011), 2174–2188.

[12] Dawes A.S., Invariant numerical methods, *Internat. J. Numer. Methods Fluids* **56** (2008), 1185–1191.

[13] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[14] Dorodnitsyn V.A. and Kozlov R., A heat transfer with a source: the complete set of invariant difference schemes, *J. Nonlin. Math. Phys.* **10** (2003), 16–50.

[15] Fels M. and Olver P.J., Moving coframes: I. A practical algorithm, *Acta Appl. Math.* **51** (1998), 161–213.

[16] Fels M. and Olver P.J., Moving coframes. II. Regularization and theoretical foundations, *Acta Appl. Math.* **55** (1999), 127–208.

[17] Fornberg B., *A practical guide to pseudospectral methods*, vol. 1 of *Cambridge Monographs on Applied and Computational Mathematics*, Cambridge University Press, Cambridge, 1996.

[18] Frank J., Gottwald G. and Reich S., A hamiltonian particle-mesh method for the rotating shallow-water equations, in *Meshfree Methods for Partial Differential Equations*, vol. 26 of *Lecture Notes in Computational Science and Engineering*, edited by M. Griebel, M.A. Schweitzer, T.J. Barth, M. Griebel, D.E. Keyes, R.M. Nieminen, D. Roose and T. Schlick, Springer, Berlin, pp. 131–142, 2002.

[19] Hairer E., Lubich C. and Wanner G., *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, Springer, Berlin, 2006.

[20] Huang D. and Zhou S., Group properties of generalized quasi-linear wave equations, *J. Math. Anal. Appl.* **366** (2010), 460–472.

[21] Huang W. and Russell R.D., *Adaptive Moving Mesh Methods*, Springer, New York, 2010.

[22] Kim P., Invariantization of numerical schemes using moving frames, *BIT Numerical Mathematics* **47** (2007), 525–546.

[23] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Phys. D* **237** (2008), 243–254.

[24] Kim P. and Olver P.J., Geometric integration via multi-space, *Regul. Chaotic Dyn.* **9** (2004), 213–226.

[25] Leimkuhler B. and Reich S., *Simulating Hamiltonian dynamics*, Cambridge University Press, Cambridge, 2004.

[26] Levi D. and Winternitz P., Continuous symmetries of difference equations, *J. Phys. A* **39** (2006), R1–R63.

[27] Olver P.J., *Application of Lie groups to differential equations*, Springer, New York, 2000.

[28] Olver P.J., Geometric foundations of numerical algorithms and symmetry, *Appl. Algebra Engrg. Comm. Comput.* **11** (2001), 417–436.

[29] Olver P.J., Generating differential invariants, *J. Math. Anal. Appl.* **333** (2007), 450–471.

[30] Ovsiannikov L.V., *Group analysis of differential equations*, Acad. Press, New York, 1982.

[31] Rebelo R. and Valiquette F., Symmetry preserving numerical schemes for partial differential equations and their numerical tests, *J. Difference Equ. Appl.* **19** (2013), arXiv:1110.5921.

[32] Sommer M. and Névir P., A conservative scheme for the shallow-water system on a staggered geodesic grid based on a Nambu representation, *Q. J. R. Meteorol. Soc.* **135** (2009), 485–494.

[33] Staniforth A. and Côté J., Semi-Lagrangian integration schemes for atmospheric models – A review, *Mon. Wea. Rev.* **119** (1991), 2206–2223.

[34] Stensrud D.J., *Parameterization schemes: Keys to understanding numerical weather prediction models*, Cambridge University Press, Cambridge, 2007.

[35] Stull R.B., *An introduction to boundary layer meteorology*, vol. 13 of *Atmospheric Sciences Library*, Kluwer Academic Publishers, Dortrecht, 1988.

[36] Valiquette F. and Winternitz P., Discretization of partial differential equations preserving their physical symmetries, *J. Phys. A* **38** (2005), 9765–9783.

Chapter 6

Convecting reference frames and invariant numerical models

Alexander Bihlo[†‡] and Jean-Christophe Nave[‡]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*

[‡] *Department of Mathematics and Statistics, McGill University, 805 Sherbrooke W., Montréal (QC) H3A 2K6, Canada*

E-mail*: bihlo@crm.umontreal.ca, jcnave@math.mcgill.ca*

In the recent paper by Bernardini et al. [*J. Comput. Phys.* **232** (2013), 1–6] the discrepancy in the performance of finite difference and spectral models for simulations of flows with a preferential direction of propagation was studied. In a simplified investigation carried out using the viscous Burgers equation the authors attributed the poorer numerical results of finite difference models to a violation of Galilean invariance in the discretization and propose to carry out the computations in a reference frame moving with the bulk velocity of the flow. Here we further discuss this problem and relate it to known results on invariant discretization schemes. Non-invariant and invariant finite difference discretizations of Burgers equation are proposed and compared with the discretization using the remedy proposed by Bernardini et al.

6.1 Introduction

In the recent paper [1] a possible remedy was discussed to improve the poor numerical behavior of finite difference simulations of turbulent flows with a preferential propagation direction. It was shown that the violation of Galilean invariance of the finite difference scheme is the most likely explanation why it is necessary to use a significantly larger number of grid points in finite difference calculations than in spectral methods to achieve comparably accurate numerical results. The recommendation given in [1] is to carry out the finite difference computations in a reference frame that moves with the constant stream-wise bulk velocity in the flow direction. It was then shown for the example of Burgers equation that the finite difference model may yield similar numerical results as spectral discretizations with approximately the same number of grid points.

In the present paper we further discuss this problem and the remedy proposed in [1]. In fact, the problem found and analyzed in [1] has been investigated quite intensively in the field of group analysis of differential and difference equations, see e.g. [3, 4, 6, 10, 12, 13, 21] and references therein for some of the most recent results. In particular, it was established by Dorodnitsyn and collaborators [6, 8, 8] that it is not possible to maintain the Galilean invariance of partial differential equations in a finite difference model when the mesh does not move in the course of the numerical integration. This result qualitatively explains why the method proposed in [1] may work from the geometrical point of view.

The violation of Galilean invariance of stationary discretizations can be readily shown by applying a Galilean boost, which in the one-dimensional case is

$$(\tilde{t}, \tilde{x}, \tilde{u}) = (t, x + \varepsilon t, u + \varepsilon), \tag{6.1}$$

where $\varepsilon \in \mathbb{R}$, to the defining equation of the grid, $x_i^{n+1} - x_i^n = 0$. Here and in the following, an upper index indicates the time level and a lower index the spatial grid point. The action of the Galilean transformation (6.1) on this grid equation yields

$$\tilde{x}_i^{n+1} - \tilde{x}_i^n = x_i^{n+1} - x_i^n + \varepsilon(t^{n+1} - t^n),$$

which clearly fails to be invariant for $\varepsilon \neq 0$. Here we assumed that all the grid points are defined on the same time layer, i.e. $t_{i+1}^n = t_i^n = t^n$. It can be checked that this assumption does not violate the invariance of most of the equations of hydrodynamics, see also [6] for more details.

Unfortunately, to maintain Galilean invariance it is also not sufficient to carry out the numerical simulations with a standard finite difference scheme in a constantly moving reference frame as proposed in [1]. It can be verified numerically that the resulting numerical solutions in the resting and in the convecting reference frames do not coincide, which is explicitly shown in Figure 6.1 for a FTCS discretization of Burgers equation. In this figure, we display the numerical solution at $t = 0.5$ in the resting reference frame (solid line) and in a reference frame which moves with constant velocity $\varepsilon_3 = 1$ (solid line with triangles) as in [1].

Instead of using a non-invariant finite difference scheme in a convecting reference frame, it is therefore desirable to construct proper finite difference discretizations that preserve the invariance group of a physical differential equations. The above observation on the incompatibility of stationary meshes with Galilean invariance have severe consequences on the design of finite difference models for the equations of fluid dynamics. In fact, it renders necessary to come up with strategies to combine the requirement of using moving meshes (in order to preserve Galilean invariance) with approaches that lead to discretization schemes having good numerical properties, such as stability, optimal grid adaptation (e.g. equidistribution of the discretization error) and the possibility for a parallel implementation. From a more general point of view, it is necessary to bridge the fields of group analysis and numerical analysis of differential equations.

To outline this connection for the example of Burgers equation considered in [1] is the main aim of the present paper. In Section 6.2 we discuss invariant finite difference schemes for Burgers equation. We construct three different types of invariant numerical schemes, namely Lagrangian discretizations, invariant adaptive Eulerian schemes and invariant schemes employing an evolution–projection strategy. We relate these schemes to the remedy for reducing the effect of violation of Galilean invariance proposed in [1]. Numerical results for the different schemes discussed are presented in Section 6.3. The final Section 6.4 contains the conclusions of the paper.

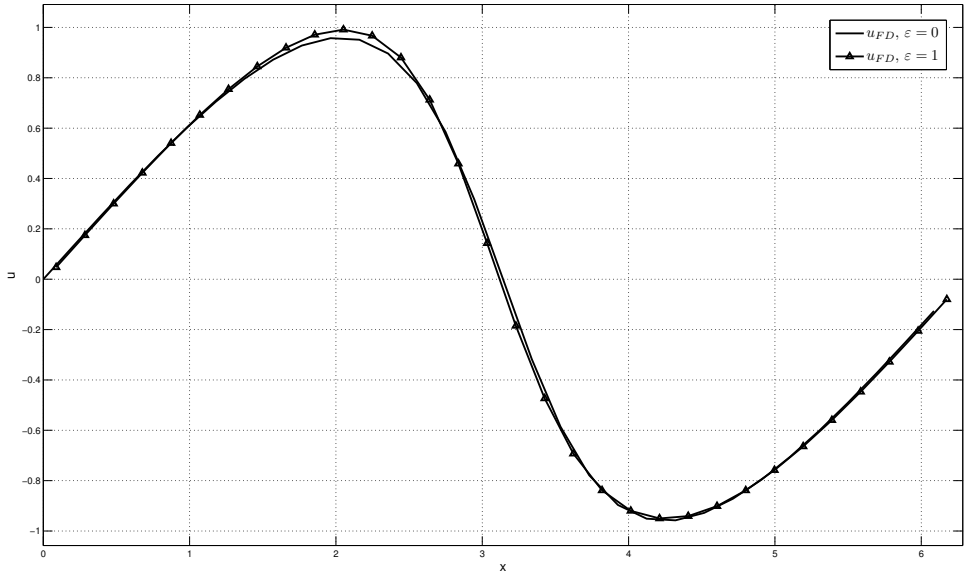

Figure 6.1: Integration using the classical FTCS discretization of Burgers equation (6.2). Solid line: Original integration in a resting reference frame. Solid lines with triangles: Integration in a reference frame moving with constant velocity $\varepsilon = 1$ as in [1]. The results in the moving reference frames were shifted back to the origin for proper comparison.

6.2 Invariant finite difference schemes for Burgers equations

As in [1], we introduce Burgers equation as a canonical example for high Reynolds number flows,

$$u_t + uu_x - \nu u_{xx} = 0, \tag{6.2}$$

where $\nu > 0$ is the viscosity, which could be scaled to 1 by means of an equivalence transformation. It is one of the most investigated models in the group analysis of differential equations, see e.g. [4, 5, 42]. Its maximal Lie invariance algebra $\mathfrak{g}$ is spanned by the basis elements

$$\partial_t, \quad \partial_x, \quad t\partial_x + \partial_u, \quad 2t\partial_t + x\partial_x - u\partial_u, \quad t^2\partial_t + tx\partial_x + (x - tu)\partial_u. \tag{6.3}$$

The associated one-parameter Lie symmetry groups are

$$\begin{aligned}
&\Gamma_1\colon \quad (t,x,u) \mapsto (t+\varepsilon_1, x, u),\\
&\Gamma_2\colon \quad (t,x,u) \mapsto (t, x+\varepsilon_2, u),\\
&\Gamma_3\colon \quad (t,x,u) \mapsto (t, x+\varepsilon_3 t, u+\varepsilon_3),\\
&\Gamma_4\colon \quad (t,x,u) \mapsto (e^{2\varepsilon_4}t, e^{\varepsilon_4}x, e^{-\varepsilon_4}u),\\
&\Gamma_5\colon \quad (t,x,u) \mapsto \left(\frac{t}{1-\varepsilon_5 t}, \frac{x}{1-\varepsilon_5 t}, u(1-\varepsilon_5 t) + \varepsilon_5 x\right),
\end{aligned} \tag{6.4}$$

showing that Burgers equation (6.2) admits time translations, space translations, Galilean boosts, scalings and time inversions as one-parameter symmetry transformations.

Invariant numerical schemes for Eq. (6.2) have already been investigated in the literature [6, 9, 10, 52]. The schemes constructed in these references preserve the entire five-parameter symmetry group G of Burgers equation. However, as was discussed in [4], it is more natural to preserve only those symmetries that are compatible with a particular set of initial and boundary value

problems chosen. In the present, we focus on periodic boundary conditions. The time inversion Γ_5 is then not compatible with a periodic domain as it does not map any periodic function u to another periodic function for $\varepsilon_5 \neq 0$. As a result we only aim to numerically preserve the first four symmetry transformations Γ_1–Γ_4. These transformations form the subgroup G^1 of the maximal Lie invariance group G of Burgers equation. It is also important to note that the subgroup G^1 is typical for various models of fluid mechanics. Consequently, the strategies discussed below are also relevant for physically more interesting higher-dimensional models of hydrodynamics, such as the Euler or Navier–Stokes equations.

Arguably, the most important observation established in the field of invariant finite difference schemes is that it is generally not possible to maintain all symmetries of a system of differential equations if the discretization scheme is constructed on a fixed, orthogonal discretization mesh [6, 13, 21]. In the case of Burgers equation, it is the presence of the Galilean transformations Γ_3 that prohibits the use of a fixed discretization mesh. This was explicitly shown in the introduction. Hence, finite difference models operating on a fixed mesh cannot be Galilean invariant.

A possible remedy is to use the following expression as a discretization of Eq. (6.2)

$$\frac{u_i^{n+1}-u_i^n}{\Delta t}+\left(u_i^n-\dot{x}_i^d\right)\frac{u_{i+1}^n-u_{i-1}^n}{x_{i+1}^n-x_{i-1}^n}-\frac{2\nu}{x_{i+1}^n-x_{i-1}^n}\left(\frac{u_{i+1}^n-u_i^n}{x_{i+1}^n-x_i^n}-\frac{u_i^n-u_{i-1}^n}{x_i^n-x_{i-1}^n}\right)=0 \tag{6.5}$$

where $\dot{x}_i^d=(x_i^{n+1}-x_i^n)/\Delta t$. Applying the transformations Γ_1–Γ_4 one readily verifies the invariance of this discretization. The reason for this scheme being invariant is that the introduced *grid velocity* $\dot{x}_i^d$ transforms as $\dot{x}_i^d \to \dot{x}_i^d+\varepsilon_4$ under the action of the Galilean transformation and the additional term involving ε_4 is exactly compensated by the Galilean transformation of u_i^n. In other words, introducing a moving mesh into the discretization of Burgers equation restores the Galilean invariance of the finite difference scheme. See [6, 12, 17, 21] for further details on the systematic construction of invariant finite difference discretization schemes.

In [4] it was shown that the above discretization can be interpreted as a discretization of (6.2) in terms of *computational coordinates*, i.e. in a coordinate system that remains fixed in the presence of grid adaptation. To accomplish this transformation, one sets $x=x(\theta,\xi)$, where $\theta=t$ and $\xi=\xi(t,x)$ is the spatial computational coordinate. Transforming (6.2) to the (θ,ξ) coordinates and discretizing the result using a FCTS scheme leads to (6.5).

The question remaining is how to determine the grid velocity $\dot{x}_i^d$, which involves the yet unknown location of the grid points on the subsequent time layer $n+1$. As pointed out in [21], there are two main strategies to find x_i^{n+1}: The first is to construct a grid equation that is invariant under the same symmetry group as the discretization of the physical differential equation. The second method is to regard the grid adaptation as unconstrained from the symmetry requirements imposed by the physical differential equation, i.e. to use a non-invariant grid equation. We will mostly focus on the first method here as it is geometrically more grounded.

In the first method we require the grid equation to be invariant under the same symmetry group as is the discretization of the physical differential equation. In the present case, this amounts to constructing a grid equation that is invariant under the transformations Γ_1–Γ_4. One simple possibility is to take

$$\dot{x}_i^d-u_i^n=0, \tag{6.6}$$

as this equation is obviously Galilean invariant and also does not violate the remaining transformations from G^1. This choice boils down to discretizing Burgers equation in *Lagrangian*

coordinates, i.e. the grid velocity equals the physical velocity. Indeed, Lagrangian discretization schemes are among the earliest examples of invariant discretizations for hydrodynamical equations, see e.g. [6]. The problem with Lagrangian discretizations is that one in general does not have proper control over the evolution of the grid points. This can be a severe problem, especially in the multi-dimensional case, where grid points can concentrate in certain regions, deteriorating the local resolution of the scheme in areas away from these concentration regions.

Perhaps numerically more satisfying are grid equations that couple the evolution of the grid to the development of pronounced features in the numerical solution, i.e. to use proper grid adaptation strategies. Linking grid adaptation to the construction of invariant discretization schemes proved relevant in the numerical investigation of blow-up problems, see e.g. [7, 8, 11]. A possible way to realize an *invariant adaptive grid* is based on the equidistribution principle for a monitor function ρ,

$$(\rho x_\xi)_\xi = 0, \tag{6.7}$$

which plays a central role in the construction of r-adaptive numerical schemes in one space dimension. See again [8, 11] and references therein for an extensive discussion of the concept of equidistributing meshes. An invariant equidistributing mesh is obtained by discretizing (6.7) in a G^1-invariant way. The general feasibility of this approach depends on the structure of the symmetry group one aims to preserve [4] but in general it can be realized for the symmetry groups one usually encounters in hydrodynamics. The basis for this approach is to choose a proper monitor function ρ, that leads to a form of (6.7) that is invariant under the symmetry subgroup G^1 of Burgers equation and then to discretize this expression in an invariant way. A G^1-invariant monitor function is

$$\rho = \sqrt{1 + \alpha u_x^2},$$

which coincides with the arc-length function for $\alpha = 1$. The reason for including a generic α in this expression is that the term u_x is not scale invariant, i.e. it transforms as $\tilde{u}_{\tilde{x}} = e^{-2\varepsilon_3} u_x$ and thus, as it stands, ρ is not scale invariant. Extending the scalings of (t, x, u) to an equivalence transformation involving α by adopting the transformation rule $\tilde{\alpha} = e^{2\varepsilon_3}\alpha$ then indeed leads to a G^1-invariant function ρ. It should be stressed though that no such extension of the Galilean transformation to an equivalence transformation is needed to guarantee the Galilean invariance of the resulting form of the equidistribution principle (6.7).

Using the modified arc-length weight function, a possible G^1-invariant discretization of (6.7) is

$$\begin{aligned} &(\rho_{i+1}^n + \rho_i^n)(x_{i+1}^{n+1} - x_i^{n+1}) - (\rho_i^n + \rho_{i-1}^n)(x_i^{n+1} - x_{i-1}^{n+1}) = 0, \\ &\rho_i^n = \sqrt{1 + \alpha \frac{u_{i+1}^n - u_{i-1}^n}{x_{i+1}^n - x_{i-1}^n}}, \end{aligned} \tag{6.8}$$

which can be solved using a relaxation scheme, such as e.g. Gauß-Seidel iteration to obtain x_i^{n+1} and hence to complete the invariant numerical scheme (6.5). More sophisticated ways to solve (6.7) are conceivable as well and could be used to improve the quality of the resulting adaptive discretization scheme.

A further possibility for the construction of invariant numerical schemes is to invoke an *evolution–projection strategy*. This idea was put forward in [40, 50] for the non-invariant discretization of advection equations and extended in [3] to find an invariant discretization of the

linear heat equation. The main approach in the evolution–projection strategy is to use the invariant numerical scheme and the invariant mesh equation only for a single integration step and to use a projection operator (i.e. an interpolation) to map the solution from the off-grid points back to the initial, uniformly spaced mesh. If the interpolation is done in an invariant way, i.e. the interpolation used preserves the invariance (sub)group of the system of differential equations being discretized, then the entire discretization procedure becomes invariant. The advantage of this approach is that moving meshes can be completely avoided.

In the present case of Burgers equation (6.2) we observe that classical interpolation schemes such as linear, quadratic or cubic spline interpolation already preserve the invariance subgroup G^1. This means that we can use the aforementioned interpolations to re-map the solution u_i^{n+1} defined at the points x_i^{n+1} back to $\hat{x}_i^{n+1} \in \{x_i^n\}$, without breaking the invariance of the scheme.

We show this explicitly for linear interpolation here, which is defined as

$$u(\hat{x}_i^{n+1}) = u_i^{n+1} + \frac{u_{i+1}^{n+1} - u_i^{n+1}}{x_{i+1}^{n+1} - x_i^{n+1}}(\hat{x}_i^{n+1} - x_i^{n+1}) = \mathcal{L}(x_i^{n+1}, x_{i+1}^{n+1}, u_i^{n+1}, u_{i+1}^{n+1}; \hat{x}_i^{n+1})$$

for the interpolation of values $\hat{x}_i^{n+1}$ lying within the interval $[x_i^{n+1}, x_{i+1}^{n+1}]$. Then, for transformations of the form Γ_1–Γ_4, we obtain that

$$\tilde{u}(\widetilde{\hat{x}}_i^{n+1}) - \mathcal{L}(\tilde{x}_i^{n+1}, \tilde{x}_{i+1}^{n+1}, \tilde{u}_i^{n+1}, \tilde{u}_{i+1}^{n+1}; \widetilde{\hat{x}}_i^{n+1}) = u(\hat{x}_i^{n+1}) - \mathcal{L}(x_i^{n+1}, x_{i+1}^{n+1}, u_i^{n+1}, u_{i+1}^{n+1}; \hat{x}_i^{n+1}),$$

which proves the invariance of linear interpolation under transformations from G^1. Similarly, invariance of quadratic and cubic spline interpolation can be shown.

Regarding the use of non-invariant grid equations, which is the second possibility to complete the description of the scheme (6.5), in principle all choices excluding $x_i^{n+1} = x_i^n$ are admissible. We will now return to the remedy proposed in [1]. The proposed approach falls into the category of non-invariant grid equations. In that paper, it was suggested to use a reference frame moving with the (constant) bulk velocity of the flow. In the case of Burgers equation, the authors set

$$x_i^{n+1} = x_i^n + c\Delta t, \tag{6.9}$$

where $c = \text{const}$. It is readily verified that this grid equation is *not* Galilean invariant, as

$$\tilde{x}_i^{n+1} - \tilde{x}_i^n - c\Delta\tilde{t} = x_i^{n+1} - x_i^n - (c - \varepsilon_4)\Delta t.$$

If $c \neq 0$ is the bulk velocity of a flow, one may indeed expect that the numerical results obtained from scheme (6.5) with grid equation (6.9) are better than for scheme (6.5) on a stationary grid. More precisely, *Galilean invariance could be restored by extending the transformation* Γ_4 *to* c *by setting* $\tilde{c} = c + \varepsilon_4$. This extension is justified in case c is related to u, which is the main reason why the remedy proposed in [1] may work. In other words, reference frames moving with a constant velocity could be made Galilean invariant in the sense that Galilean transformations have an extension to equivalence transformations for such reference frames. A similar extension of the transformation Γ_3 to c is necessary to incorporate the scale invariance.

For the sake of convenience, we summarized the characteristics of the different schemes discussed in the present section in Table 6.1.

6.3 Numerical results

In this section we present some numerical results obtained from the invariant discretization schemes introduced in the previous section. For all the experiments, we use $u(0, x) = \sin(x)$

Table 6.1: Different numerical schemes for the viscous Burgers equation (6.2).

Numerical scheme	Grid equation	Galilean invariance	Grid spacing
Finite differences	None	Not invariant	constant
Lagrangian	Lagrangian grid movement	fully invariant	variable
Eulerian adaptive	Equidistribution principle	fully invariant	variable
[1] Bernardini et al.	Constant grid movement	semi-invariant	constant
Evolution–projection	Lagrangian grid movement	fully invariant	constant

as the initial condition on a 2π-periodic domain and the integration is carried out up to time $t = 0.5$ for $\nu = 0.1$. Unless otherwise stated, we use $N = 64$ grid points and fix the time step with $\Delta t \propto h^2$, where h is the mean grid spacing over the domain.

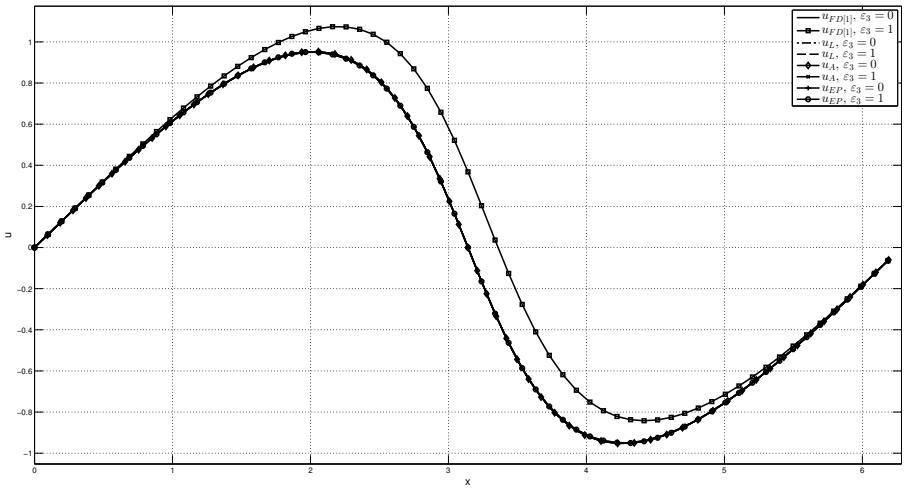

Figure 6.2: Integration of Burgers equation (6.2) using the three invariant discretization schemes introduced in Section 6.2 as well as the scheme presented in [1]. *Solid line:* Finite differences in a resting reference frame as in [1]. *Solid line with squares:* Finite differences in a constantly moving reference frame as in [1], $\varepsilon_3 = 1$, $c = -0.5$. *Dashed–dotted line:* Lagrangian scheme in the resting reference frame. *Dashed line:* Lagrangian scheme in a reference frame moving with constant velocity $\varepsilon_3 = 1$. *Solid line with diamonds:* Eulerian adaptive scheme in the resting reference frame. *Solid line with crosses:* Eulerian adaptive scheme in a reference frame moving with constant velocity $\varepsilon_3 = 1$. *Solid line with pluses:* Evolution–projection scheme in the resting reference frame. *Solid line with circles:* Evolution–projection scheme in a reference frame moving with constant velocity $\varepsilon_3 = 1$.

In Figure 6.2 we carry out numerical integrations for three invariant numerical schemes for Burgers equation based on (6.5) and employing different grid equations: (i) The fully Lagrangian scheme uses the Lagrangian grid equation (6.6), (ii) in the invariant Eulerian adaptive scheme the grid points $\{x_i^{n+1}\}$ is determined from the invariant discretization of the equidistribution principle (6.7) and (iii) in the evolution–projection scheme we use the Lagrangian grid equation (6.6) and use quadratic interpolation to re-map the off-grid points to their original location at the previous time step. For the sake of reference, we also add the scheme proposed in [1].

For all schemes we numerically verify Galilean invariance, i.e. each of the pairs of integrations in a resting and a constantly moving coordinate system yields visually the same numerical

solution for (i) the Lagrangian scheme (dashed–dotted line, $\varepsilon_3 = 0$ and dashed line, $\varepsilon_3 = 1$), (ii) the Eulerian adaptive scheme (solid line with diamonds, $\varepsilon_3 = 0$ and solid line with crosses $\varepsilon_3 = 1$) and (iii) the evolution–projection scheme (solid line with pluses, $\varepsilon_3 = 0$ and solid line with circles $\varepsilon_3 = 1$). Moreover, it is seen from Figure 6.2 that all three schemes yield approximately the same numerical solution. In contrary, the scheme proposed in [1] is not invariant (solid line, $\varepsilon = 0$ and solid line with squares $\varepsilon = 1$ and $c = -0.5$), illustrating the discussion from Section 6.1.

We aim to detail and analyze this result further now.

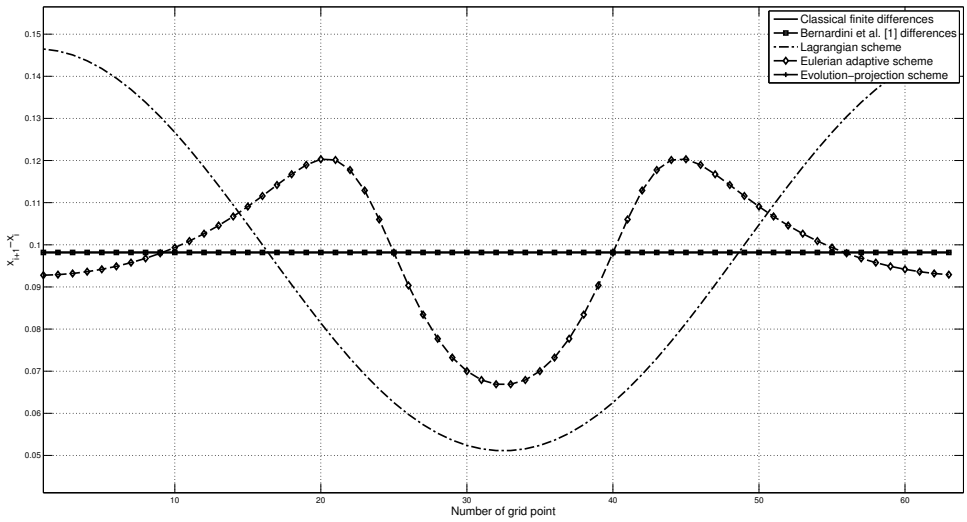

Figure 6.3: Grid spacing $\Delta x_i^n = x_{i+1}^n - x_i^n$ at final time $t = 0.5$ for the five discretizations of Burgers equation shown in Figures 6.1 and 6.2.

The difference between the three invariant numerical schemes for Burgers equation is the invoked grid equation. For the sake of a clearer presentation, we depict the grid spacing $\Delta x_i^n = x_{i+1}^n - x_i^n$ as a function of the location of the grid points x_i^n at the final integration time $t = 0.5$ in Figure 6.3. For the classical, non-invariant finite difference scheme the spacing is by definition constant (solid line) as is in the scheme proposed in [1] (solid line with squares). For the Lagrangian discretization (dashed line) the location of the grid points depends on the solution itself and therefore one does not have control over the local resolution. It is a mere consequence of equating the physical velocity and the grid velocity. In the adaptive Eulerian scheme (dashed line with diamonds) we observe a proper concentration of the grid points along the building shock, which follows from using the equidistribution principle and the arc-length monitor function. Away from the steepening front, the points remain quasi-equally distributed. By construction, the grid points in the evolution–projection method (solid line with pluses) are again equally spaced.

To estimate the overall accuracy of the different schemes, in Figure 6.4 we display the pointwise differences of the numerical solutions and the exact solution of the chosen initial value problem $u(0, x) = \sin(x)$ for Burgers equation, which is

$$u_e(t,x) = 2\nu \frac{\sum_{j=1}^{\infty} a_j j e^{-\nu t j^2} \sin jx}{\sum_{j=0}^{\infty} a_j e^{-\nu t j^2} \cos jx},$$

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} e^{-(1-\cos x)/(2\nu)} \mathrm{d}x, \quad a_{j>0} = \frac{1}{\pi} \int_0^{2\pi} e^{-(1-\cos x)/(2\nu)} \cos jx \, \mathrm{d}x.$$

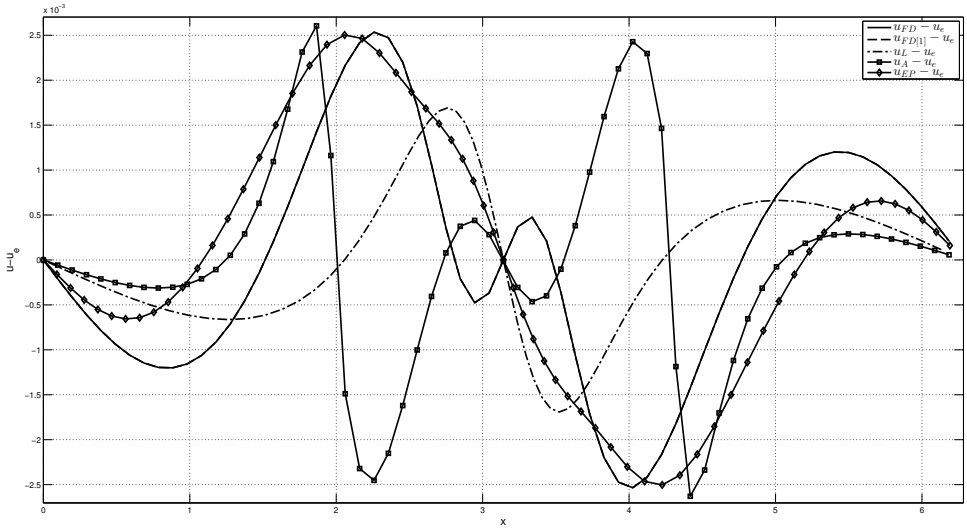

Figure 6.4: Pointwise difference of the numerical and the exact solution at final time $t = 0.5$ for the five discretizations of Burgers equation shown in Figures 6.1 and 6.2.

It can be seen from Figure 6.4 that all the numerical schemes achieve a comparable accuracy (with the scheme proposed in [1] coinciding with classical finite differences). The overall l_∞-errors of the schemes for the runs depicted in Figure 6.4 are summarized in Table 6.2.

Table 6.2: l_∞-errors for the various invariant discretization schemes for the viscous Burgers equation (6.2) with $N = 64$ grid points. Classical finite differences are included as reference.

	classical FD	Lagrangian	Eulerian adaptive	Evolution–projection
$\|\|E\|\|_{l_\infty}$	$2.53 \cdot 10^{-3}$	$1.69 \cdot 10^{-3}$	$2.50 \cdot 10^{-3}$	$2.63 \cdot 10^{-3}$

The convergence rates of the different schemes in the l_∞-norm are depicted in Figure 6.5 using $N \in \{4, 8, 16, 32, 64, 128, 256, 512\}$ grid points. It is seen from this plot that all schemes yield comparable errors. The overall convergence rates demonstrate that all three types of invariant schemes for Burgers equation introduced in this paper are asymptotically of second order and that the different invariant discretization strategies do not alter the accuracy of the underlying approximations. More details on this can be found in [3].

6.4 Conclusion

In the present paper we have revisited the problem recently pointed out in [1] that classical finite difference discretizations of the governing equations of hydrodynamics violate Galilean invariance. We have discussed three possible ways of constructing Galilean invariant finite difference schemes for Burgers equation: (i) Lagrangian discretization schemes, (ii) Eulerian adaptive discretizations and (iii) discretizations using an evolution–projection strategy. These three approaches can be readily adapted to the two- and three-dimensional Euler or Navier–Stokes equations. The approach proposed in [1] leads to a semi-invariant scheme in that the authors use the discretization on a moving coordinate system but employ a non-invariant grid equation by moving the grid points with a constant velocity (e.g. the bulk velocity) rather than with the actual physical velocity. If properly done, this approach can indeed reduce the effects

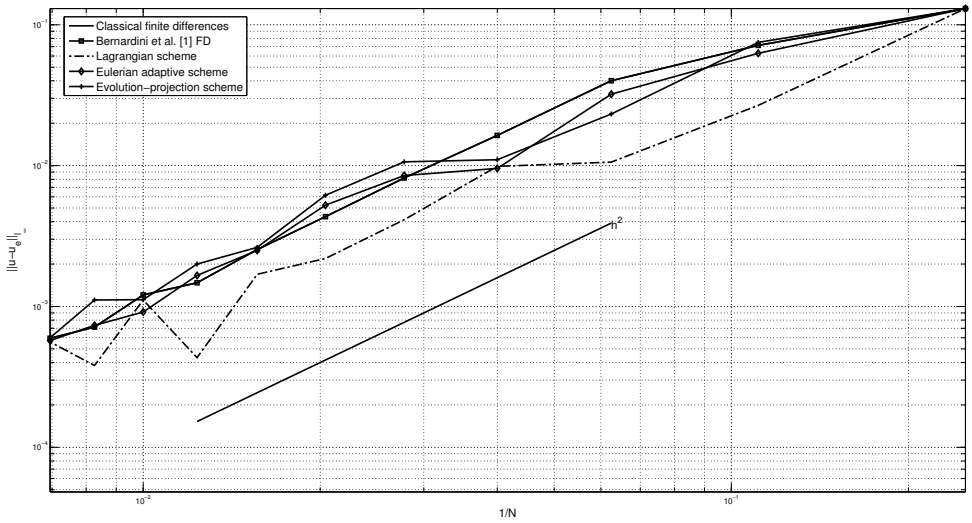

Figure 6.5: Convergence plots for the five numerical schemes presented above for $N \in \{4, 8, 16, 32, 64, 128, 256, 512\}$ grid points. *Solid line:* Classical finite difference scheme. *Solid line with squares:* Scheme [1]. *Dashed–dotted line:* Lagrangian scheme. *Solid line with diamonds:* Eulerian adaptive scheme. *Solid line with pluses:* Evolution–projection scheme.

of the violation of Galilean invariance in classical finite difference discretizations, while still not yielding a fully Galilean invariant scheme.

All of the invariant discretization approaches presented above have their advantages and disadvantages. Purely Lagrangian discretization schemes are not in widespread use as such schemes usually lead to a strong concentration of grid points, leaving other regions of the domain poorly resolved. Moreover, in multi-dimensional cases of interest in hydrodynamics, Lagrangian schemes can lead to tangled meshes. At the same time, Lagrangian schemes are good in that they are able to preserve sharp interfaces within a fluid. The Eulerian adaptive approach is attractive because it provides a natural way to link the problem of finding invariant discretization schemes to the properties of the numerical solution at each time step. At the same time, the computational overhead required to efficiently generate the meshes can be a crucial factor determining the feasibility of the adaptation methodology, especially for multi-dimensional systems. Finally, in situations where adaptive numerical schemes are not desirable, the evolution–projection strategy based on the Lagrangian grid equation (or any other invariant grid equation one is able to find) and a simple (but invariant) interpolation is a possible way to maintain Galilean invariance in a finite difference scheme while still being able to operate on a fixed, uniformly spaced mesh. The drawback of the evolution–projection approach is that it also requires an additional operation, the interpolation, which may cause additional computational overhead compared to, for instance, the Lagrangian scheme.

Applied to Burgers equation, the three invariant discretization methodologies yielded numerical schemes that are asymptotically second order accurate. Compared to the standard FTCS scheme, which is also second order accurate, one thus gets as a bonus the preservation of a geometric feature of Burgers equation, namely a subgroup of its symmetry group. The preservation of this symmetry subgroup can be a crucial factor for problems for which it is inevitable to carry out the simulations in a moving reference frame. In fact, the benefits offered by an invariant scheme compared to its non-invariant counterpart of the same order is difficult to assess. On one

hand, conservation properties of invariant schemes only become evident when long integration times are considered. On the other hand, in the case of turbulent channel flows, having a high order numerical scheme guarantees that small features present in the flow are well accounted for. Only the combination of both properties can lead to accurate long time integration of such systems. As a result, the construction of higher order yet invariant numerical schemes is a subject of great importance in both technique and application. This is the subject of current and future research.

Acknowledgements

This research was supported by the Austrian Science Fund (FWF), project J3182–N13 (AB). JCN wishes to acknowledge partial support from the NSERC Discovery and Discovery Accelerator Programs. The authors thank the anonymous referee for helpful remarks and suggestions.

Bibliography

[1] Bernardini M., Pirozzoli S., Quadrio M. and Orlandi P., Turbulent channel flow simulations in convecting reference frames, *J. Comput. Phys.* **232** (2013), 1–6.

[2] Bihlo A. and Nave J.-C., Invariant discretization schemes for the heat equation, *SIGMA* **9**, 052, 23 pp, arXiv:1209.5028.

[3] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[4] Bluman G. and Kumei S., *Symmetries and differential equations*, Springer, New York, 1989.

[5] Bluman G.W., Cheviakov A.F. and Anco S.C., *Application of symmetry methods to partial differential equations*, Springer, New York, 2010.

[6] Budd C. and Dorodnitsyn V.A., Symmetry-adapted moving mesh schemes for the nonlinear Schrödinger equation, *J. Phys. A* **34** (2001), 10387–10400.

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

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

[9] Chhay M. and Hamdouni A., A new construction for invariant numerical schemes using moving frames, *C. R. Mecanique* **338** (2010), 97–101.

[10] Chhay M., Hoarau E., Hamdouni A. and Sagaut P., Comparison of some Lie-symmetry-based integrators, *J. Comput. Phys.* **230** (2011), 2174–2188.

[11] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[12] Dorodnitsyn V.A. and Kozlov R., A heat transfer with a source: the complete set of invariant difference schemes, *J. Nonlin. Math. Phys.* **10** (2003), 16–50.

[13] Huang W. and Russell R.D., *Adaptive Moving Mesh Methods*, Springer, New York, 2010.

[14] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Physica D* **237** (2008), 243–254.

[15] Levi D. and Winternitz P., Continuous symmetries of difference equations, *J. Phys. A* **39** (2006), R1–R63.

[16] Nave J.-C., Rosales R.R. and Seibold B., A gradient-augmented level set method with an optimally local, coherent advection scheme, *J. Comput. Phys.* **229** (2010), 3802–3827.

[17] Olver P.J., *Application of Lie groups to differential equations*, Springer, New York, 2000.

[18] Olver P.J., Geometric foundations of numerical algorithms and symmetry, *Appl. Algebra Engrg. Comm. Comput.* **11** (2001), 417–436.

[19] Rebelo R. and Valiquette F., Symmetry preserving numerical schemes for partial differential equations and their numerical tests, *J. Difference Equ. Appl.* **19** (2013), 738–757, arXiv:1110.5921, 2012.

[20] Seibold B., Rosales R.R. and Nave J.-C., Jet schemes for advection problems, *Discrete Contin. Dyn. Syst. Ser. B* **17** (2012), 1229–1259.

[21] Valiquette F. and Winternitz P., Discretization of partial differential equations preserving their physical symmetries, *J. Phys. A* **38** (2005), 9765–9783.

Chapter 7

The Korteweg–de Vries equation and its symmetry-preserving discretization

Alexander Bihlo[†], Xavier Coiteux-Roy[‡] and Pavel Winternitz[§]

[†] *Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's (NL), A1C 5S7, Canada*

[‡] *Département de physique and Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*

[§] *Département de mathématiques et de statistique and Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*[1]

E-mail*: abihlo@mun.ca, xavier.coiteux-roy@umontreal.ca, wintern@crm.umontreal.ca*

The Korteweg–de Vries equation is one of the most important nonlinear evolution equations in the mathematical sciences. In this article invariant discretization schemes are constructed for this equation both in the Lagrangian and in the Eulerian form. We also propose invariant schemes that preserve the momentum. Numerical tests are carried out for all invariant discretization schemes and related to standard numerical schemes. We find that the invariant discretization schemes give generally the same level of accuracy as the standard schemes with the added benefit of preserving Galilean transformations which is demonstrated numerically as well.

7.1 Introduction

This article is part of a general program the purpose of which is to study the possibility of discretizing the equations of physics while preserving their Lie point symmetries [6–8, 13, 14, 16–18, 21, 34–37, 49, 53, 54]. There are both conceptual and practical reasons for doing this. From the conceptual point of view symmetries under rotations, Galilei or Lorentz transformations, conformal and other transformations are of primordial importance both in classical and quantum

[1] Presently on sabbatical leave at: Dipartimento di Matematica e Fisica, Università degli Studi Roma Tre, Via della Vasca Navale 84, 00146 Roma, Italy

physics. Indeed, in fields such as nuclear and particle physics where the dynamical laws are not well established, symmetries provide crucial restrictions on the form of the interaction. For instance, they determine the possible general form of the nucleon-nucleon potential [41, 45]. It would be a pity to lose such symmetries and their implications in the study of physics in a discrete world. From the practical point of view symmetries of differential equations determine many of the properties of solutions. Preserving symmetries in a discretization should provide difference systems that share some exact solutions with the original differential equations, or at least provide better approximations than noninvariant systems. In turn, this should have implications for numerical solutions. Thus, symmetry preserving discretizations should provide solutions that are in some sense "better" than "standard" discretizations.

The basic idea [6, 13, 53] of this approach is to approximate a differential equation by a "difference system" consisting of several discrete equations. The solutions of this system determine the lattice and approximate the solution of the differential equation. In the continuous limit the lattice equations reduce to identities (like $0 = 0$) and the remaining equations go to the appropriate differential equation. The difference scheme is constructed out of invariants of the Lie point symmetry group G of the differential equation. The action of G on the independent and dependent variables is the same as for the continuous case and this action is assumed to be known. The action of G is not prolonged to derivatives, but to all points of the lattice (the "discrete jet space").

This invariant discretization approach has been extensively applied to ordinary differential equation (ODEs). It has been shown that for first order ODEs an invariant discretization is exact [49]. The solution of an invariant difference scheme coincides point by point with the appropriate solution of the ODE. Moreover it is sufficient if the difference system is invariant under a one-dimensional subgroup of the symmetry group.

For second and third order ODEs it is often possible to integrate the invariant scheme directly and thus see explicitly how solutions of the difference scheme converge to those of the ODE [16–18, 54]. It has been shown on the example of numerous second and third order nonlinear ODEs that the invariant discretizations provide more accurate numerical solutions than standard methods [6, 7, 48]. This is specially so in the neighborhood of singularities where invariant methods, as opposed to standard ones, make it possible to continue solutions beyond the singularities.

For partial differential equations (PDEs) the first application of Lie group theory to numerical methods is, to our knowledge, due to Shokin and Yanenko [51, 55]. Their approach "Differential approximation" is quite different from ours (for a comparison see [36]).

Quite a few articles devoted to the symmetry adapted discretization of PDEs have appeared over the last 20 years (see e.g. [2–4, 4, 7, 8, 8–10, 12, 17, 21, 21, 31, 35, 47, 52]). Invariant discretizations of the Korteweg–de Vries (KdV) equation were presented in [6, 14, 52].

The purpose of this article is to study invariant discretizations of the KdV equations in greater depth. Thus we will compare the known invariant discretizations amongst each other and propose new ones. All of them will be tested as numerical schemes for known exact solutions. Their accuracy and stability will be evaluated by comparing with known analytic solutions.

The KdV equation is very suitable for such a study. On one hand, it is an integrable equation so a very large body of analytical solutions is known (due to inverse scattering techniques [1, 24]). On the other hand the KdV equation has an interesting Lie point symmetry group that includes Galilei invariance. It is a prototype of a Galilei invariant evolution equation that can

be invariantly discretized on a mesh with horizontal time lines, but not on an orthogonal one (nor any other equally spaced one).

The original invariant discretizations [7, 14] essentially correspond to using the Lagrange formulation of hydrodynamics in the continuous limit. We suggest an alternative discretization that is natural in the Eulerian formalism, especially when combined with adaptive computational schemes.

In Section 7.2 we review some well known results on the symmetry group of the continuous KdV equation and on its known analytical solutions. We also present the Lagrangian form of the KdV equation. The invariant discretizations are presented in Section 7.3. All numerical results are concentrated in Section 7.4. The final Section 7.5 is devoted to the conclusions.

7.2 The continuous KdV equation

We shall write the KdV equation in the form

$$u_t + uu_x + u_{xxx} = 0. \tag{7.1}$$

Its Lie point symmetry group is well-known (see e.g. [42]). A basis for its Lie algebra $\mathfrak{g}$ is given by the vector fields

$$\mathcal{D} = 3t\partial_t + x\partial_x - 2u\partial_u, \quad \mathcal{B} = t\partial_x + \partial_u, \quad \mathcal{P}_1 = \partial_x, \quad \mathcal{P}_0 = \partial_t, \tag{7.2}$$

corresponding to dilations, Galilei boosts and space and time translations, respectively.

The symmetry algebra $\mathfrak{g}$ has precisely five conjugacy classes of one-dimensional subalgebras. A representative list of these classes is given by the algebras

$$\{\mathcal{D}\}, \quad \{\mathcal{B}\}, \quad \{\mathcal{B} + \mathcal{P}_0\}, \quad \{\mathcal{P}_0\}, \quad \{\mathcal{P}_1\}. \tag{7.3}$$

Conjugacy is considered under the group of inner automorphisms of (7.1), extended by the simultaneous reflections of x and t

$$\mathrm{R}x = -x, \quad \mathrm{R}t = -t, \quad \mathrm{R}u = u. \tag{7.4}$$

Thus, $G = \mathrm{R} \Ni G_0$, where $G_0 = e^{d\mathcal{D}} e^{v\mathcal{B}} e^{t_0\mathcal{P}_0} e^{x_0\mathcal{P}_1}$.

The group can be used to get new solutions from known ones. If $u(t, x)$ is a solution of the KdV equation then so are $u(-t, -x)$ and

$$\tilde{u}(\tilde{t}, \tilde{x}) = e^{2d} u\left(e^{-3d}(t - t_0), e^{-d}(x - x_0 - v(t - t_0))\right) + e^{-2d} v, \qquad d, v, t_0, x_0 \in \mathbb{R}, \tag{7.5}$$

where d, v, t_0 and x_0 are group parameters.

7.2.1 Lagrangian formulation of the KdV equation

The original form of the KdV equation (7.1) is written in *Eulerian variables*, i.e. the velocity u is a function of time and space, $u = u(t, x)$. An alternative to the Eulerian form is the *Lagrangian form*. In the Lagrangian description of fluid mechanics the velocity u is a function of time and of the original position of the *fluid particle* ξ. Assuming that the fluid particles maintain their identity (hence ξ is independent of time), one needs to express the KdV equation as an

equation for $u = u(\tau, x(\tau, \xi))$, where $\tau = t$. Using the chain rule, the Eulerian form of the KdV equation (7.1) is transformed to

$$u_\tau + (u - x_\tau)\frac{u_\xi}{x_\xi} + \frac{1}{x_\xi}\left(\frac{1}{x_\xi}\left(\frac{u_\xi}{x_\xi}\right)_\xi\right)_\xi = 0. \tag{7.6}$$

Up to now, no particular relation between the original physical coordinate x and the new Lagrangian coordinate ξ has been imposed. In the classical Lagrangian framework, this change of coordinates is specified by setting

$$x_\tau(\tau, \xi) = u(x(\tau, \xi), t). \tag{7.7}$$

In other words, the change of variables from the Lagrangian coordinates to the Eulerian coordinates is completed upon integrating the equation for the particle trajectories (7.7). The KdV equation in Lagrangian coordinates then reduces to

$$u_\tau + \frac{1}{x_\xi}\left(\frac{1}{x_\xi}\left(\frac{u_\xi}{x_\xi}\right)_\xi\right)_\xi = 0.$$

The change of coordinates from the Eulerian form (7.1) to the form (7.6) is more general than the particular Lagrangian case given through (7.7). In the more general case, the variables (τ, ξ) are referred to as the *computational* coordinates. From the numerical point of view, using the KdV equation in computational coordinates (7.6) gives the perspective of defining the relation $x = x(\tau, \xi)$ in such a manner that the evolution of the discretization grid is coupled to the evolution of the KdV equation itself. This is the main idea of using adaptive numerical schemes [11]. The importance of such schemes in the framework of invariant discretization will be clarified in Section 7.3.

We should like to stress here that even for the more general form (7.6) of the KdV equation with yet unspecified relation $x = x(\tau, \xi)$ it makes sense to fix the transformation $\tau = t$. This guarantees that the resulting equation will be of evolutionary type (though it would be sufficient to put $\tau = \tau(t)$).

7.2.2 Symmetry reduction and exact solutions

One of the reasons why exact analytical solutions of PDEs are useful is that they can be used to check the accuracy of numerical algorithms, in particular the invariant discretizations to be presented below. For integrable equations with nontrivial symmetry groups (like the KdV equation) there exist two main sources of exact solutions. One is symmetry reduction, producing solutions invariant under some subgroup of the symmetry group. The other is the method of inverse scattering and its generalizations that lead to multisoliton and periodic and quasiperiodic solutions.

Let us start with the method of symmetry reduction. In order to reduce the KdV equation to an ODE we impose that the solution $u(t, x)$ be invariant under a one-dimensional subgroup $of G_0$ corresponding to a one-dimensional subalgebra of the symmetry algebra $\mathfrak{g}$. The classification of these subalgebras leads to the list (7.3). Invariance under a subgroup corresponding to the algebra element $X = \tau\partial_t + \zeta\partial_x + \phi\partial_u$ corresponds to imposing that $u(t, x)$ in addition to (7.1) should satisfy the quasilinear first order PDE

$$\tau u_t + \zeta u_x = \phi. \tag{7.8}$$

This equation is solved and the result is put into the KdV equation (7.1) which reduces to an ODE.

Let us run through the individual subalgbreas listed in (7.3).

(i) $\mathcal{P}_1 = \partial_x$. From (7.8) we obtain $u = f(t)$ and (7.1) implies that

$$u(t,x) = A. \tag{7.9}$$

Thus, the only solution of the KdV invariant under space translations is a constant.

(ii) $\mathcal{B} = t\partial_x + \partial_u$. From (7.8) we get the reduction formula

$$u(t,x) = \frac{x}{t} + f(t).$$

Substituting into (7.1) and solving the obtained ODE for $f(t)$ we find $f(t) = \frac{A}{t}$. Applying the group transformations (7.5) we obtain the Galilei (and dilation) invariant solution

$$u(t,x) = \frac{x - x_0}{t - t_0}. \tag{7.10}$$

(ii) $\mathcal{B} + \mathcal{P}_0 = t\partial_x + \partial_u + \partial_t$. The reduction formula following from (7.8) is

$$u(t,x) = t + f(\gamma), \quad \gamma = x - \frac{1}{2}t^2.$$

The KdV equation reduces to $f''' + ff' + 1 = 0$. Integrating once and putting

$$f(\gamma) = -12^{3/5} P\left[\left(\frac{1}{12}\right)^{1/5}(\gamma) + \delta\right]$$

we find that $P(z)$ satisfies the first Painlevé equation

$$P'' = 6P^2 + z, \tag{7.11}$$

see [25, 28]. The corresponding solution of the KdV equation is

$$u(x,t) = t - 12^{3/5} P_{\mathrm{I}}\left[\left(\frac{1}{12}\right)^{1/5}(\gamma) + \delta\right],$$

where P_{I} is the first Painlevé transcendent and δ is an arbitrary constant. No elementary solutions of (7.11) are known.

(iv) $\mathcal{D} = 3t\partial_t + x\partial_x - 2u\partial_u$. The reduction formula (7.8) in this case yields

$$u = t^{-2/3} F(\gamma), \quad \gamma = xt^{-1/3},$$

where $F(\gamma)$ satisfies

$$F''' + FF' - \frac{1}{3}\gamma F' - \frac{2}{3}F = 0. \tag{7.12}$$

The Miura transformation [42] $F = w' - w^2/6$ and subsequent integration takes (7.12) into

$$w_{\gamma\gamma} = \frac{1}{18}w^3 + \frac{1}{3}\gamma w + k. \tag{7.13}$$

Eq. (7.13) can be reduced to the equation

$$P'' = 2P^3 + zP + \alpha, \tag{7.14}$$

where α is an arbitrary constant. This is the equation for the second Painlevé transcendent $P_{\rm II}$. Finally, the dilationally invariant solution of the KdV equation is

$$u_\alpha(t,x) = 2(3)^{1/3}t^{-2/3}(P''_\alpha(\gamma) - P^2_\alpha(\gamma)), \quad \gamma = xt^{-1/3}, \tag{7.15}$$

where P_α is a solution of (7.14) (we have denoted P_α the $P_{\rm II}$ transcendent viewed as a function of the parameter α). Contrary to $P_{\rm I}$, the $P_{\rm II}$ equation allows two families of elementary solutions for special values of the parameter α [25]. For integer values $\alpha = \pm n$ these are rational solutions. For half integer values $\alpha = \pm(2n+1)/2$ the solutions are expressed in terms of Airy functions. In both cases they satisfy $P_\alpha = -P_{-\alpha}$ and are listed in [25] for low values of n. For the combination $W_\alpha = P''_\alpha - P^2_\alpha$ we observe an additional relation, namely $W_{n+1} = -W_{-n}$, so for convenience we restrict to $\alpha = 0, -1, -2, -3$. We thus obtain the following dilationally invariant solutions of the KdV equation

$$\begin{aligned} &u_0 = 0, \quad , u_{-1} = -\frac{12}{x^2}, \quad u_{-2} = -\frac{36x(24t - x^3)}{(12t + x^3)^2}, \\ &u_{-3} = -\frac{72(x^9 + 5400x^3t^2 + 43200t^2)x}{(720t^2 - 60x^3t - x^6)^2}. \end{aligned} \tag{7.16}$$

The solution u_0 is also invariant under space and time translations, u_{-1} is also invariant under time translations.

(v) $\boldsymbol{\mathcal{P}_0 = \partial_t}$. Solutions invariant under time translations have the form $u = f(x)$. A Galilei transformations boosts such a solution to a traveling wave $u = f(x - \lambda t) + \lambda$.

Substituting into the KdV equation and integrating twice we get an ODE that can be written as

$$(f')^2 = -\frac{1}{3}(f-a)(f-b)(f-c), \quad a + b + c = 0. \tag{7.17}$$

The roots of the polynomial in (7.17) can all be real. Then we order them to have $a \geq b \geq c$. The other possibility is $a \in \mathbb{R}$, $b = \bar{c} = p + iq$, with $q > 0$, $p, q \in \mathbb{R}$.

We are interested in real solutions only. They may be finite or singular (for $x \in \mathbb{R}$), periodic or localized. Let us run through the individual cases. Solutions are expressed in terms of Jacobi elliptic functions [11] or degenerate cases thereof.

Cnoidal waves: $c < b \leq f \leq a$, $b < a$. The solution in this case reads

$$u(t,x) = b + (a-b)\mathrm{cn}^2(\omega x, k), \quad k = \sqrt{\frac{a-b}{2a+b}}, \quad \omega = \sqrt{\frac{2a+b}{3}}, \quad 2a + b > 0. \tag{7.18}$$

We can apply a Galilei boost with $v = -b$ and obtain the more usual form

$$u(t,x) = (a+v)\mathrm{cn}^2(\omega(x - vt), k), \quad k = \sqrt{\frac{a+v}{2a-v}}, \quad \omega = \sqrt{\frac{2a-v}{3}}. \tag{7.19}$$

Soliton: $c = b \leq f \leq a$, $b = -\frac{a}{2}$, $k = 1$, $\omega = \frac{1}{2}\sqrt{\frac{a}{2}}$. The associated solution of the KdV equation is

$$u(t,x) = -\frac{a}{2} + \frac{3a}{2}\frac{1}{\cosh^2 \frac{1}{2}\sqrt{\frac{a}{2}}x}, \quad a > 0. \tag{7.20}$$

After a boost with $a = 2v$ we have the usual KdV soliton

$$u(t,x) = \frac{3v}{\cosh^2 \frac{1}{2}\sqrt{v}(x-vt)}. \tag{7.21}$$

Singular snoidal solution: $f \le c < b < a$. The solution of the KdV equation reads

$$u(t,x) = a - \frac{a-c}{\mathrm{sn}^2(\omega x, k)}, \quad \omega = \frac{1}{2}\sqrt{\frac{a-c}{3}}, \quad k = \sqrt{\frac{2a+c}{a-c}}. \tag{7.22}$$

Singular soliton: $f \le c = b < a$. The solution of the KdV equation in this case is

$$u = -\frac{a}{2}\left(1 + \frac{3}{\sinh^2(\omega x)}\right), \quad \omega = \frac{1}{2}\sqrt{\frac{a}{2}}. \tag{7.23}$$

Singular trigonometric solution: $f \le c < b = a$. We obtain the solution

$$u = a - \frac{3a}{\sin^2(\omega x)}, \quad \omega = \frac{1}{2}\sqrt{a}. \tag{7.24}$$

Singular algebraic soliton: $a = b = c = 0$. The solution of the KdV equation in this case reduces to

$$u = -\frac{12}{x^2}, \tag{7.25}$$

which coincides with the solution u_{-1} listed in (7.16) which is thus invariant under dilations and time translations. Galilei transformations take it into

$$u(t,x) = -\frac{12}{(x-vt)^2} + v. \tag{7.26}$$

Real solutions corresponding to complex roots: $f \le a \in \mathbb{R}$, $b = -\frac{a}{2} + iq$, $\quad c = -\frac{a}{2} - iq$, $q > 0$. The corresponding solution of the KdV equation is

$$u(t,x) = a - A\frac{1+\mathrm{cn}(\omega x,k)}{1-\mathrm{cn}(\omega x,k)}, \quad A = \sqrt{\frac{9a^2}{4}+q^2}, \quad \omega = \sqrt{\frac{A}{3}}, \quad k^2 = \frac{(A+\frac{3a}{2})^2+q^2}{4A^2}. \tag{7.27}$$

An elementary special case is obtained for $k = 1$, i.e. $a = \pm\frac{2}{3}$, $A = \sqrt{1+q^2}$, namely

$$u(t,x) = \pm\frac{2}{3} - \sqrt{1+q^2} - \frac{\sqrt{1+q^2}}{\sinh^2 \frac{\omega x}{2}}, \quad \omega = \sqrt{\frac{1+q^2}{3}}. \tag{7.28}$$

Other exact solutions are obtained by the inverse scattering method [1, 24]. Amonst them the most relevant for this article is the double soliton

$$\begin{aligned} &u(t,x) = 12\frac{\partial^2}{\partial x^2}\ln(1 + B_1 e^{iQ_1} + B_2 e^{iQ_2} + AB_1B_2e^{i(Q_1+Q_2)}),\\ &Q_1 = a_1x - a_1^3t, \quad a_2x - a_2^3t, \quad A = \left(\frac{a_1-a_2}{a_1+a_2}\right)^2, \end{aligned} \tag{7.29}$$

where a_1, a_2, B_1 and B_2 are arbitrary constants. Real solutions are obtained by putting $a_1 = i\alpha_1$, $a_2 = i\alpha_2$ with $\alpha_1, \alpha_2, B_1, B_2 \in \mathbb{R}$.

Many other solutions (n-soliton, multigap quasiperiodic solutions, etc.) are available in the literature [20, 22, 23, 25, 32].

7.3 Invariant discretization of the KdV equation

7.3.1 Invariant discretization on a ten point stencil

The KdV equation is a scalar (1+1)-dimensional evolution equation. In the finite difference approximation on the t-x-plane, the continuous space of independent variables (t, x) is sampled by a collection of finite points $\{P_i^n\}$ only. Here and in the following, we use the double index notation (t_i^n, x_i^n) to denote a discrete point in this t-x-plane, where $i \in \mathbb{Z}$ is the spatial index and $n \in \mathbb{N}$ is the temporal index. Likewise, the dependent functions are defined on the associated points $\{P_i^n\}$ only, i.e. $u_i^n = u(t_i^n, x_i^n)$.

A partial differential equation $\mathcal{L}\colon \Delta(x, u^{(q)}) = 0$, where $u^{(q)}$ denotes all the derivatives of u with respect to t and x up to order q, is discretized in a symmetry-preserving manner if it is expressed by a consistent finite difference approximation that can be written as a function of the finite difference invariants of the symmetry group of the equation itself. By *consistent* it is meant that in the continuous limit (i.e. the distance between the points $\{P_i^n\}$ goes to zero) the finite difference approximation converges to the original differential equation $\mathcal{L}$.

In writing this discretization, it is not only necessary to define a finite difference approximation of the differential equation $\mathcal{L}$ itself but also to specify the lattice of points $\{P_i^n\}$ in an invariant fashion. In other words, the equation $\mathcal{L}$ is replaced by a system of finite difference equations of the form

$$\Delta S\colon E_\alpha(t_i^n, x_i^n, u_i^n) = 0, \quad \alpha = 1, \dots, N, \quad i_{\min} \le i \le i_{\max}, \quad 0 \le n \le n_{\max}$$

where the number of equations N in the system ΔS is at least $N = 3$.

The general method for finding invariant numerical schemes using difference invariants can be found e.g. in [6, 13]. Here we only present the respective computations for the KdV equation. We should also like to mention here that there is another method for finding invariant discretization schemes that rests on invariantization using *equivariant moving frames*. For more information on this alternative method, see e.g. [3, 12, 17, 21].

The minimum number of points in the stencil to discretize the derivatives in the KdV equation is five as spatial derivatives up to order three and a first order time derivative have to be approximated. In order to increase the accuracy of the finite difference approximation we introduce an extended ten point stencil. Lower order approximations can be obtained by restricting oneself to a subset of these 10 stencil points.

The stencils used are depicted in Fig. 7.3.1. It can be seen that a two-step time integration is employed allowing for either forward Euler (six point stencil, squares), backward Euler (six point stencil, crosses) or trapezoidal time integrators (ten point stencil, solid circles). Invariant numerical schemes using higher order time-stepping are possible as well but will not be presented here.

To simplify the notation, we also introduce the following abbreviations

$$\Delta\tau = t^{n+1} - t^n, \qquad h_i^n = x_{i+1}^n - x_i^n, \qquad Du_i^n = \frac{u_{i+1}^n - u_i^n}{h_i^n},$$

for the spacings and elementary first order discrete derivatives. Note that the spacing in time does not carry an index as we use equally spaced, horizontal time layers only. It is readily checked that variable time-stepping would leave the following numerical scheme invariant as well, as long as the time-step control is invariant itself. See the similar discussion for the spatial adaptation strategies presented in Section 7.3.4.

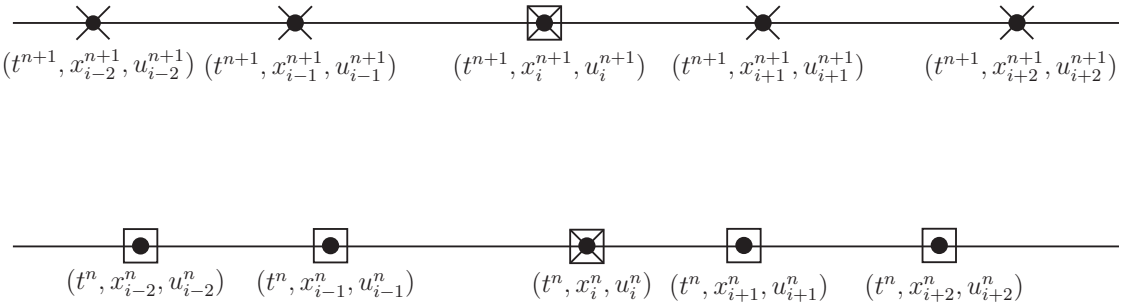

Figure 7.1: Stencils for the discretization of the KdV equation: Ten point stencil (solid circles). Explicit six point stencil (squares). Implicit six point stencil (crosses).

The prolongation of vector fields of the maximal Lie invariance algebra $\mathfrak{g}$ to the stencil shown in Fig. 7.3.1 yields

$$
\begin{aligned}
&\partial_{t^n} + \partial_{t^{n+1}},\\
&\partial_{x^n_i} + \partial_{x^n_{i+1}} + \partial_{x^n_{i+2}} + \partial_{x^n_{i-1}} + \partial_{x^n_{i-2}} + \partial_{x^{n+1}_i} + \partial_{x^{n+1}_{i+1}} + \partial_{x^{n+1}_{i+2}} + \partial_{x^{n+1}_{i-1}} + \partial_{x^{n+1}_{i-2}},\\
&t^n(\partial_{x^n_i} + \partial_{x^n_{i+1}} + \partial_{x^n_{i+2}} + \partial_{x^n_{i-1}} + \partial_{x^n_{i-2}}) + t^{n+1}(\partial_{x^{n+1}_i} + \partial_{x^{n+1}_{i+1}} + \partial_{x^{n+1}_{i+2}} + \partial_{x^{n+1}_{i-1}} + \partial_{x^{n+1}_{i-2}})\\
&\quad + \partial_{u^n_i} + \partial_{u^n_{i+1}} + \partial_{u^n_{i+2}} + \partial_{u^n_{i-1}} + \partial_{u^n_{i-2}} + \partial_{u^{n+1}_i} + \partial_{u^{n+1}_{i+1}} + \partial_{u^{n+1}_{i+2}} + \partial_{u^{n+1}_{i-1}} + \partial_{u^{n+1}_{i-2}},\\
&x^{n+1}_i\partial_{x^{n+1}_i} + x^{n+1}_{i+1}\partial_{x^{n+1}_{i+1}} + x^{n+1}_{i+2}\partial_{x^{n+1}_{i+2}} + x^{n+1}_{i-1}\partial_{x^{n+1}_{i-1}} + x^{n+1}_{i-2}\partial_{x^{n+1}_{i-2}}\\
&\quad + 3(t^{n+1}_i\partial_{t^{n+1}_i} + t^{n+1}_{i+1}\partial_{t^{n+1}_{i+1}} + t^{n+1}_{i+2}\partial_{t^{n+1}_{i+2}} + t^{n+1}_{i-1}\partial_{t^{n+1}_{i-1}} + t^{n+1}_{i-2}\partial_{t^{n+1}_{i-2}})\\
&\quad - 2(u^{n+1}_i\partial_{u^{n+1}_i} + u^{n+1}_{i+1}\partial_{u^{n+1}_{i+1}} + u^{n+1}_{i+2}\partial_{u^{n+1}_{i+2}} + u^{n+1}_{i-1}\partial_{u^{n+1}_{i-1}} + u^{n+1}_{i-2}\partial_{u^{n+1}_{i-2}}).
\end{aligned}
\tag{7.30}
$$

A complete list of functionally independent finite difference invariants annihilated by the prolonged infinitesimal generators on the ten point stencil (7.30) is exhausted by

$$
\begin{aligned}
&I_1 = \frac{h^n_{i-1}}{h^n_i}, \quad I_2 = \frac{h^n_{i+1}}{h^n_i}, \quad I_3 = \frac{h^n_{i-2}}{h^n_i}, \quad I_4 = \frac{h^{n+1}_i}{h^n_i}, \quad I_5 = \frac{h^{n+1}_{i-1}}{h^n_i}, \quad I_6 = \frac{h^{n+1}_{i+1}}{h^n_i},\\
&I_7 = \frac{h^{n+1}_{i-2}}{h^n_i}, \quad I_8 = \frac{(h^n_i)^3}{\Delta\tau}, \quad I_9 = \frac{x^{n+1}_i - x^n_i - \tau u^n_i}{h^n_i}, \quad I_{10} = (u^{n+1}_i - u^n_i)(h^n_i)^2,\\
&I_{11} = \Delta\tau D u^n_i, \quad I_{12} = \Delta\tau D u^n_{i+1}, \quad I_{13} = \Delta\tau D u^n_{i-1}, \quad I_{14} = \Delta\tau D u^n_{i-2},\\
&I_{15} = \Delta\tau D u^{n+1}_i, \quad I_{16} = \Delta\tau D u^{n+1}_{i+1}, \quad I_{17} = \Delta\tau D u^{n+1}_{i-1}, \quad I_{18} = \Delta\tau D u^{n+1}_{i-2}.
\end{aligned}
\tag{7.31}
$$

Building the numerical scheme for the KdV equation and the lattice using these invariants guarantees that the resulting scheme is invariant under the same maximal Lie invariance group G as is the KdV equation. We first start with the discretization of (7.6).

It turns out that the straightforward discretization of the KdV equation in terms of the computational coordinates (τ, ξ) given by (7.6) is already invariant under the maximal Lie invariance group G. We demonstrate this first for the explicit six point stencil scheme here. Indeed, the invariant finite difference expression,

$$
I_{10} - I_8 I_9 \frac{I_{11} + I_{13}}{2} + \frac{1}{2}\left[2\frac{I_{12} - I_{11}}{1 + I_2} - 2\frac{I_{11} - I_{13}}{1 + I_1} + \frac{1}{I_1}\left(2\frac{I_{11} - I_{13}}{1 + I_1} - 2\frac{I_{13} - I_{14}}{I_1 + I_3}\right)\right] = 0,
$$

reads explicitly

$$\frac{u_i^{n+1}-u_i^n}{\Delta\tau}+(u_i^n-\dot{x}_i)\frac{Du_i^n+Du_{i-1}^n}{2}+\frac{1}{2h_i^n}\left[\frac{2\left(Du_{i+1}^n-Du_i^n\right)}{h_{i+1}^n+h_i^n}-\frac{2\left(Du_i^n-Du_{i-1}^n\right)}{h_i^n+h_{i-1}^n}\right]$$
$$+\frac{1}{2h_{i-1}^n}\left[\frac{2\left(Du_i^n-Du_{i-1}^n\right)}{h_i^n+h_{i-1}^n}-\frac{2\left(Du_{i-1}^n-Du_{i-2}^n\right)}{h_{i-1}^n+h_{i-2}^n}\right]=0, \tag{7.32}$$

after some re-arrangements, where

$$\dot{x}_i=\frac{x_i^{n+1}-x_i^n}{\Delta\tau},$$

denotes the *grid velocity*. Correspondingly, this discretization preserves the four-dimensional maximal Lie invariance group of the KdV equation. In order to use the scheme (7.32) it is necessary to specify an invariant equation for the grid velocity. This will be pursued in the following subsections.

The continuous limit of scheme (7.32) is taken by parameterizing the spacings h_i^n as a function of computational coordinates ξ. This implies that

$$h_i^n=x_\xi\Delta\xi$$

and a Taylor series expansion of (7.32) gives that

$$u_\tau+(u-x_\tau)\frac{u_\xi}{x_\xi}+\frac{1}{x_\xi}\left(\frac{1}{x_\xi}\left(\frac{u_\xi}{x_\xi}\right)_\xi\right)_\xi=O(\Delta\tau,\Delta\xi^2).$$

Thus, as expected, the scheme (7.32) is of first order in time and second order in space. From the numerical point of view the scheme (7.32) is not advantageous as the forward in time discretization is unconditionally unstable.

A more appropriate numerical scheme can be realized on the entire ten point lattice and is given by

$$I_{10}-I_8I_9\frac{I_{11}+I_{13}+I_{15}+I_{17}}{4}+\frac{1}{4}\Bigg[\left(2\frac{I_{16}-I_{15}}{I_4+I_6}-2\frac{I_{15}-I_{17}}{I_4+I_5}\right)$$
$$+\frac{1}{I_5}\left(2\frac{I_{15}-I_{17}}{I_4+I_5}-2\frac{I_{17}-I_{18}}{I_5+I_7}\right)$$
$$+\left(2\frac{I_{12}-I_{11}}{1+I_2}-2\frac{I_{11}-I_{13}}{1+I_1}\right)+\frac{1}{I_1}\left(2\frac{I_{11}-I_{13}}{1+I_1}-2\frac{I_{13}-I_{14}}{I_1+I_3}\right)\Bigg]=0,$$

which reads in expanded form as

$$\frac{\hat{u}-u}{\Delta\tau}+(u_i^n-\dot{x}_i)\frac{Du_i^n+Du_{i-1}^n+Du_i^{n+1}+Du_{i-1}^{n+1}}{4}$$
$$+\frac{1}{4h_i^{n+1}}\left[\frac{2\left(Du_{i+1}^{n+1}-Du_i^{n+1}\right)}{h_{i+1}^{n+1}+h_i^{n+1}}-\frac{2\left(Du_i^{n+1}-Du_{i-1}^{n+1}\right)}{h_i^{n+1}+h_{i-1}^{n+1}}\right]$$
$$+\frac{1}{4h_{i-1}^{n+1}}\left[\frac{2\left(Du_i^{n+1}-Du_{i-1}^{n+1}\right)}{h_i^{n+1}+h_{i-1}^{n+1}}-\frac{2\left(Du_{i-1}^{n+1}-Du_{i-2}^{n+1}\right)}{h_{i-1}^{n+1}+h_{i-2}^{n+1}}\right] \tag{7.33}$$
$$+\frac{1}{4h_i^n}\left[\frac{2\left(Du_{i+1}^n-Du_i^n\right)}{h_{i+1}^n+h_i^n}-\frac{2\left(Du_i^n-Du_{i-1}^n\right)}{h_i^n+h_{i-1}^n}\right]$$
$$+\frac{1}{4h_{i-1}^n}\left[\frac{2\left(Du_i^n-Du_{i-1}^n\right)}{h_i^n+h_{i-1}^n}-\frac{2\left(Du_{i-1}^n-Du_{i-2}^n\right)}{h_{i-1}^n+h_{i-2}^n}\right]=0.$$

In the continuous limit, this scheme becomes

$$u_\tau + (u - x_\tau)\frac{u_\xi}{x_\xi} + \frac{1}{x_\xi}\left(\frac{1}{x_\xi}\left(\frac{u_\xi}{x_\xi}\right)_\xi\right)_\xi = O(\Delta\tau, \Delta\xi^2).$$

which is still of first order in time due to the particular way the grid velocity has been discretized. Due to the use of the trapezoidal rule, the resulting scheme is conditionally stable now. The implicit six point stencil scheme is constructed in a similar fashion.

For the sake of reference we also present the standard forward in time, centered in space scheme on an orthogonal and stationary six point lattice for the KdV equation expressed in Eulerian form (7.1) here:

$$h_i^n = h = \text{const}, \quad \Delta\tau = \Delta t,$$
$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + u_i^n \frac{u_{i+1}^n - u_{i-1}^n}{2h} + \frac{u_{i+2}^n - 2u_{i+1}^n + 2u_{i-1}^n - u_{i-2}^n}{2h^3} = 0,$$

It is readily checked that this discretization scheme breaks the Galilean invariance of the KdV equation while preserving invariance under both shifts and dilations. The standard, non-invariant implicit schemes on the six and ten point stencils are defined in a similar manner but not given here.

7.3.2 Invariant Lagrangian discretization schemes

In order to complete the numerical scheme (7.32) and (7.33) it is necessary to formulate an equation for the grid velocity. In the purely Lagrangian scheme one uses the discretization of the relation (7.7), which is

$$\frac{x_i^{n+1} - x_i^n}{\Delta\tau} = u_i^n. \tag{7.34}$$

That is, the grid velocity coincides with the physical velocity. It is well known that a purely Lagrangian scheme can perform poorly as there is no built-in mechanism preventing the clustering of grid points as the numerical integration proceeds [11]. In the higher-dimensional case, usually mesh tangling occurs when using Lagrangian schemes.

An alternative to using (7.34) to obtain the position of the grid points on the next time level is to use *adaptive moving mesh methods*. These will be shortly introduced in Section 7.3.4.

7.3.3 Invariant evolution–projection discretization

A possibility to make invariant Lagrangian schemes numerically competitive is to invoke them in an evolution–projection strategy [40, 50]. The main idea is to use the invariant Lagrangian scheme introduced in the previous subsection only for a single time step and then project the solution defined on the new grid points $\{x_i^{n+1}\}$ back to the original grid $\{x_i^n\}$. This way, mesh movement can be effectively avoided. The projection step is in general accomplished through interpolation and the invariance of the whole solution procedure is guaranteed if the interpolation method used is invariant under the same symmetry group that has been used to construct the numerical scheme itself. This strategy has been successfully adapted for the linear heat equation and the viscous Burgers equation [3, 4].

We show here that polynomial interpolation of any order is invariant under the maximal Lie invariance group of the KdV equation and hence can be used in an invariant evolution–projection

scheme for this equation. In the numerical results below we then choose quadratic interpolation as using it in conjunction with a second order invariant numerical scheme guarantees that the whole evolution–projection procedure is second order accurate. However, standard higher order interpolation could be used as well in invariant evolution–projection schemes for the KdV equation.

As our goal is to interpolate the solution u_i^{n+1} defined at time t^{n+1} back to the grid as given on time level t^n the appropriate form of the mth order polynomial interpolation formula is

$$u^{n+1}(x) = \sum_{i=0}^{m} L_i(x) u_i^{n+1}, \tag{7.35}$$

where

$$L_i(x) = \prod_{\substack{0 \leqslant j \leqslant m \\ j \neq i}} \frac{x - x_j^{n+1}}{x_i^{n+1} - x_j^{n+1}}$$

are the Lagrange polynomials and $x \in [x_0^{n+1}, x_m^{n+1}]$ is the point where the solution $u^{n+1}(x)$ should be interpolated. It is readily seen that the interpolation formula (7.35) is invariant under space and time translations as well as under the scale symmetry of the KdV equation. Galilean invariance $(\widetilde{t^n}, \widetilde{x_i^n}, \widetilde{u_i^n}) = (t^n, x_i^n + \varepsilon t^n, u_i^n + \varepsilon)$ is respected by (7.35) too, as

$$\widetilde{u^{n+1}}(x) = u^{n+1}(x) + \varepsilon = \sum_{i=0}^{m} \widetilde{L_i}(x) \widetilde{u_i^{n+1}} = \sum_{i=0}^{m} L_i(x)(u_i^{n+1} + \varepsilon) = \left(\sum_{i=0}^{m} L_i(x) u_i^{n+1} \right) + \varepsilon$$

thus leading back to (7.35). Note that we have used here the property of the Lagrange polynomials that

$$\sum_{i=0}^{m} L_i(x) = 1.$$

Specifying the general polynomial interpolation (7.35) to quadratic interpolation for the KdV equation on the ten point stencil can be done e.g. by setting $(x_0^{n+1}, u_0^{n+1}) = (x_{i-2}^{n+1}, u_{i-2}^{n+1})$, $(x_1^{n+1}, u_1^{n+1}) = (x_i^{n+1}, u_i^{n+1})$ and $(x_2^{n+1}, u_2^{n+1}) = (x_{i+2}^{n+1}, u_{i+2}^{n+1})$. In practice, the projection step is completed by choosing the interpolating point $x \in \{x_i^n\}$, i.e. by evaluating the solution $u^{n+1}(x)$ at the location of the old grid points.

7.3.4 Invariant adaptive discretization schemes

Before we give the form of an invariant adaptive scheme for the KdV equation we introduce some basic background material related to adaptive numerical schemes in general. More information can be found, e.g. in the textbook [11].

Adaptive discretization schemes

The main idea behind moving mesh methods is to link the evolution of a mesh to the numerical solution of the discretized PDE itself. In the case of a Lagrangian scheme the new location of the grid points is determined by the solution u itself only. A better criterion is usually to link the evolution of the grid points to the derivatives of u. This can be accomplished through the computation of *equidistributing meshes*.

Definition 7.1. Let $\rho(x)$ be a strictly positive continuous function on the interval $[a,b]$. Let $a = x_1 < x_2 < \cdots < x_{N-1} < x_N = b$ be a partition (i.e. a mesh) of this interval. The mesh is said to be equidistributing for ρ on $[a,b]$ if

$$\int_{x_1}^{x_2} \rho(x)\mathrm{d}x = \int_{x_2}^{x_3} \rho(x)\mathrm{d}x = \cdots = \int_{x_{N-1}}^{x_N} \rho(x)\mathrm{d}x \tag{7.36}$$

holds.

The function ρ is called *mesh density function* or monitor function. For the practical implementation it is advantageous to convert the relation (7.36) into a differential equation. This is done by first using the equivalent expression

$$\int_a^{x_j} \rho(x)\mathrm{d}x = \frac{(j-1)}{N-1}\int_a^b \rho(x)\mathrm{d}x = \xi_j \int_a^b \rho(x)\mathrm{d}x,$$

where ξ_j, $j = 1,\ldots,N$, is the discrete computational coordinate. By definition, $\xi_j \in [0,1]$.

Regarding x as a function of the computational coordinate, i.e. $x_j = x(\xi_j)$, in the continuous limit the above integral equation becomes

$$\int_a^{x(\xi)} \rho(x)\mathrm{d}x = \xi \int_a^b \rho(x)\mathrm{d}x,$$

which holds for all $\xi \in [0,1]$. Differentiating this equation twice with respect to ξ leads to

$$(\rho(x)x_\xi)_\xi = 0, \tag{7.37}$$

which is the differential form of the equidistribution principle when subjected to the boundary conditions $x(0) = a$ and $x(1) = b$.

So as to complete the description of a numerical scheme upon using the equidistribution principle in its differential form (7.37) one needs to specify the mesh density function ρ. A classical choice is the arc-length type function

$$\rho = \sqrt{1 + \alpha u_x^2},$$

where $\alpha \in \mathbb{R}$ is a constant parameter governing the strength of the adaptation. Other monitor functions, such as built around the curvature of u are used as well.

Invariant adaptive scheme for the KdV equation

In order to complete the invariant numerical scheme for the KdV equation one has to discretize the differential form of the equidistribution principle (7.37) in an invariant way. As the missing ingredient in the grid velocity $\dot{x}_i$ is x_i^{n+1}, we discretize (7.37) on the time layer t^{n+1}. This is done upon composing a discretization of (7.37) out of the difference invariants for the KdV equation (7.31). A possible discretization using the arc-length type mesh density function is:

$$\frac{\rho_{i+1}^n + \rho_i^n}{2} I_{11} - \frac{\rho_i^n + \rho_{i-1}^n}{2}\frac{I_{13}}{I_1} = 0,$$

where

$$\rho_{i+1} = \sqrt{1 + \alpha I_{11}^2}, \quad \rho_i = \sqrt{1 + \alpha I_{12}^2}, \quad \rho_{i-1} = \sqrt{1 + \alpha I_{13}^2},$$

or, explicitly,

$$\frac{\rho_{i+1}^n + \rho_i^n}{2}(x_{i+1}^{n+1} - x_i^{n+1}) - \frac{\rho_i^n + \rho_{i-1}^n}{2}(x_i^{n+1} - x_{i-1}^{n+1}) = 0, \tag{7.38}$$

where

$$\rho_i^n = \sqrt{1 + \alpha\left(\Delta\tau \frac{u_{i+1}^n - u_i^n}{x_{i+1}^n - x_i^n}\right)^2}. \tag{7.39}$$

7.3.5 Momentum preserving invariant discretization

It is well-known that the KdV equation admits infinitely many conservation laws, see e.g. [42] for a discussion. Numerically preserving conservation laws of partial differential equations is generally a nontrivial problem that belongs to the realm of geometric numerical integration. More information on this field can be found in the books [26, 33]. The problem of finding finite difference discretizations for the KdV equation that preserve sub-sets of the infinite span of conservation laws is a complicated problem that will not be investigated here. We are only concerned with finding invariant discretization schemes that also preserve linear momentum

$$\mathcal{M} = \int u \, \mathrm{d}x.$$

This conservation law is associated with expressing the KdV equation itself in conserved form

$$\mathrm{D}_t u + \mathrm{D}_x\left(\frac{1}{2}u^2 + u_{xx}\right) = 0.$$

It is possible to preserve the above conserved form also on a moving mesh, which as we have seen above is a basic requirement for preserving Galilean invariance. In particular, the following discretization is invariant under the maximal Lie invariance group of the KdV equation and momentum-preserving:

$$\begin{aligned}&\frac{(h_i^{n+1} + h_{i-1}^{n+1})u_i^{n+1} - (h_i^n + h_{i-1}^n)u_i^n}{\Delta\tau} - \left(\frac{x_{i+1}^{n+1} - x_{i+1}^n}{\Delta\tau}u_{i+1}^n - \frac{x_{i-1}^{n+1} - x_{i-1}^n}{\Delta\tau}u_{i-1}^n\right)\\ &+ \frac{1}{2}((u_{i+1}^n)^2 - (u_{i-1}^n)^2) + \left[\frac{2(Du_{i+1}^n - Du_i^n)}{h_{i+1}^n + h_i^n} - \frac{2(Du_{i-1}^n - Du_{i-2}^n)}{h_{i-1}^n + h_{i-2}^n}\right] = 0.\end{aligned} \tag{7.40}$$

The associated continuous expression to this discretization is

$$(x_\xi u)_\tau + \left(\frac{1}{2}u^2 + \left(\frac{1}{x_\xi}\left(\frac{u_\xi}{x_\xi}\right)_\xi\right)_\xi - ux_\tau\right)_\xi = 0,$$

which is of conserved form in the computational coordinates. It thus discretely conserves momentum $\mathcal{M}$.

Let us now show that (7.40) also preserves all the Lie symmetries as admitted by the KdV equation. One way of showing this would be to express (7.40) in terms of the difference invariants (7.31). However, due to the particular form of (7.40) a direct expression in terms of difference invariants would be cumbersome. It is much easier to verify invariance directly by transforming the scheme (7.40) under the action of the symmetry group of the KdV equation.

It is obvious that the discretization (7.40) is invariant under shifts in space and time as well as under scale transformations. It thus only remains to show invariance under Galilean transformations $(t^n, x_i^n, u_i^n) \mapsto (t^n, x_i^n + \varepsilon t^n, u_i^n + \varepsilon)$. We proceed term by term:

$$\frac{(\widetilde{h_i^{n+1}} + \widetilde{h_{i-1}^{n+1}})\widetilde{u_i^{n+1}} - (\widetilde{h_i^n} + \widetilde{h_{i-1}^n})\widetilde{u_i^n}}{\widetilde{\Delta\tau}} = \frac{(h_i^{n+1} + h_{i-1}^{n+1})u_i^{n+1} - (h_i^n + h_{i-1}^n)u_i^n}{\Delta\tau}$$
$$+ \varepsilon \left(\frac{x_{i+1}^{n+1} - x_{i+1}^n}{\Delta\tau} - \frac{x_{i-1}^{n+1} - x_{i-1}^n}{\Delta\tau} \right),$$
$$\left(\frac{\widetilde{x_{i+1}^{n+1}} - \widetilde{x_{i+1}^n}}{\widetilde{\Delta\tau}} \widetilde{u_{i+1}^n} - \frac{\widetilde{x_{i-1}^{n+1}} - \widetilde{x_{i-1}^n}}{\widetilde{\Delta\tau}} \widetilde{u_{i-1}^n} \right) = \left(\frac{x_{i+1}^{n+1} - x_{i+1}^n}{\Delta\tau} u_{i+1}^n - \frac{x_{i-1}^{n+1} - x_{i-1}^n}{\Delta\tau} u_{i-1}^n \right)$$
$$+ \varepsilon(u_{i+1}^n - u_{i-1}^n) + \varepsilon \left(\frac{x_{i+1}^{n+1} - x_{i+1}^n}{\Delta\tau} - \frac{x_{i-1}^{n+1} - x_{i-1}^n}{\Delta\tau} \right),$$
$$\frac{1}{2}((\widetilde{u_{i+1}^n})^2 - (\widetilde{u_{i-1}^n})^2) = \frac{1}{2}((u_{i+1}^n)^2 - (u_{i-1}^n)^2) + \varepsilon(u_{i+1}^n - u_{i-1}^n),$$
$$\left[\frac{2(\widetilde{Du_{i+1}^n} - \widetilde{Du_i^n})}{\widetilde{h_{i+1}^n} + \widetilde{h_i^n}} - \frac{2(\widetilde{Du_{i-1}^n} - \widetilde{Du_{i-2}^n})}{\widetilde{h_{i-1}^n} + \widetilde{h_{i-2}^n}} \right] =$$
$$\left[\frac{2(Du_{i+1}^n - Du_i^n)}{h_{i+1}^n + h_i^n} - \frac{2(Du_{i-1}^n - Du_{i-2}^n)}{h_{i-1}^n + h_{i-2}^n} \right]$$

Substituting into the transformed form of equation (7.40) proves Galilean invariance.

As it stands, the momentum preserving invariant scheme (7.40) still needs to be completed by adapting an appropriate strategy to obtain the new mesh $\{x_i^{n+1}\}$. Here, the same strategies as proposed above for the case of the non-conservative invariant scheme (7.33) can be applied. These strategies lead to *invariant momentum-preserving Lagrangian*, *evolution–projection* and *adaptive schemes*, respectively.

7.3.6 Exact discretization

An interesting question on the behavior of numerical schemes is whether they are able to reproduce exact solutions of the original differential equation exact, i.e. without numerical error.

Among all the exact solutions given in Section 7.2.2, the only solutions that are exact for all schemes reported in Section 7.3 is the constant solution (7.9). In addition, the Galilean invariant solution (7.10) is an exact solution for the invariant Lagrangian schemes (7.32) and (7.33) using (7.34) which is readily verified directly. Below we verify numerically that this solution is also exact for the invariant evolution–projection scheme and the invariant momentum preserving scheme.

7.4 Numerical results

In this section we collect the numerical results obtained using the various schemes proposed in the previous section. Our purpose is not to do a technical optimization of every scheme but to rather demonstrate the feasibility of implementing invariant discretization schemes as well as the resulting physical implications.

For the invariant adaptive scheme, we use the discretization (7.38) of the equidistribution principle with the invariant mesh density function (7.39). To compare the invariant adaptive scheme against a non-invariant adaptive one we also use the mesh density function $\rho = \sqrt{1 + \alpha u_{xx}^2}$, discretized by

$$\rho_i^n{}_{\text{non-inv}} = \sqrt{1 + \alpha \left(\Delta\tau \frac{2\frac{u_{i+2}^n - u_i^n}{x_{i+2}^n - x_i^n} - 2\frac{u_{i+1}^n - u_{i-1}^n}{x_{i+1}^n - x_{i-1}^n}}{x_{i+2}^n - x_i^n + x_{i+1}^n - x_{i-2}^n} \right)^2}. \tag{7.41}$$

in conjunction with (7.38). Similar mesh density functions are also used in adaptive numerical schemes, see e.g. [11]. In the present case, using (7.41) breaks the scale invariance in the discretization of the KdV equation. The resulting scheme therefore serves as reference for a non-invariant adaptive scheme.

Note that for the sake of brevity we abbreviate the standard notation $a \cdot 10^n$ in the tables and figure legends below by a*e*n.

7.4.1 Decaying cosine evolution

Before we use the exact solutions computed in Section 7.2.2 as benchmark tests, we reproduce the classical results obtained by Zabusky and Kruskal in 1965 [56] of a wave decaying into solitons. For this experiment, Zabusky and Kruskal used the following form of the KdV equation

$$u_t + uu_x + \delta^2 u_{xxx} = 0,$$

where $\delta = 0.022$. The initial condition used was $u = \cos(\pi x)$ on a periodic domain of length $L = 2$. Zabusky and Kruskal observed the formation of eight solitons at time $t = 3.6/\pi$.

A main problem reproducing this result with the invariant Lagrangian schemes (7.32) and (7.33) using (7.34) is that mesh tangling occurs before the final integration time $t = 3.6/\pi$. In turn, using the invariant Lagrangian scheme only in the framework of the invariant evolution–projection method allows us to arrive at a solution at the final integration time.

All the other schemes presented above are able to compute this test problem. The results of these integrations are shown in Fig. 7.2. From this figure it can be seen that all schemes are capable of capturing the decay into solitons as originally presented in [56]. It can also be seen that the two evolution–projection schemes show a slight lag for the first four solitons when compared to the high resolution solution. The other schemes lie visually very close to this high resolution solution.

To quantify these findings, in Table 7.1 we present the root-mean-square error (RMSE) for the various schemes tested, using the high resolution finite difference solution as reference. The RMSE is defined by

$$\text{RMSE} = \sqrt{\sum_{i=1}^{N} \frac{(u_{\text{num}_\text{i}} - u_{\text{exact}_\text{i}})^2}{N}}.$$

where in place of the exact solution, u_{exact}, the high resolution numerical solution is used.

It can be seen from this table that the evolution–projection schemes have indeed errors larger by a factor of ten than the other schemes tested, which all give quite comparable errors. A possible explanation for this increase of error is that the interpolation used does not accurately take into account the rapid change in the first derivatives of the numerical solution. Using higher order interpolation incorporating derivative information, such as Hermite interpolation, could help reduce this phase error in the evolution–projection scheme, see also [40, 50].

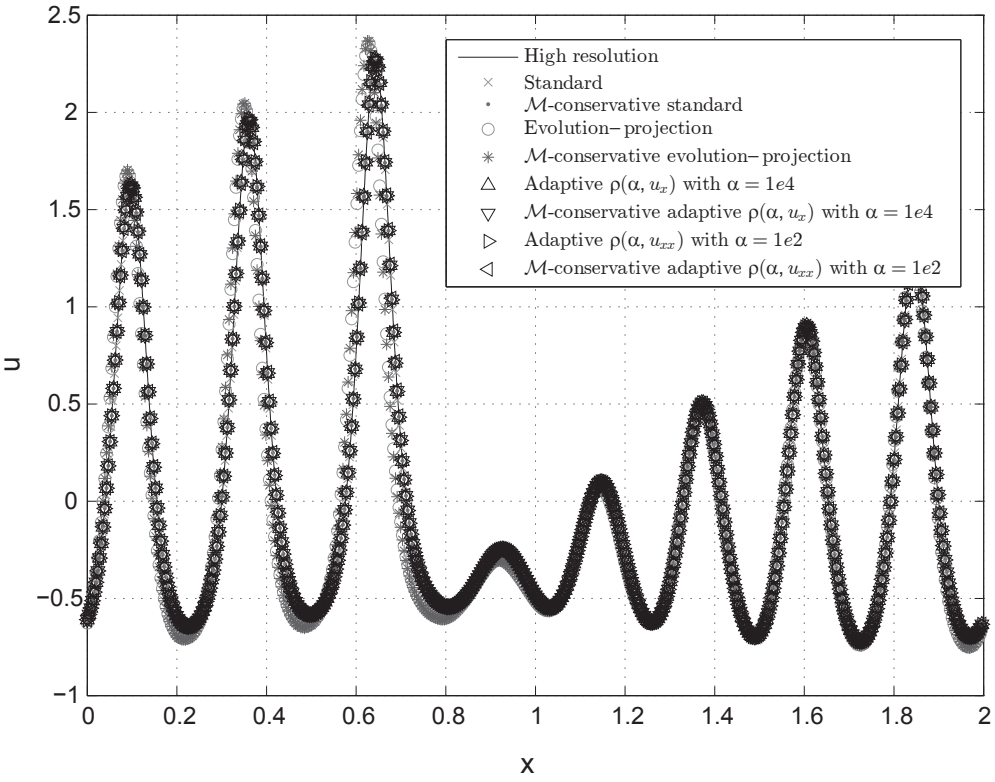

Figure 7.2: Numerical solution of the Zabusky–Kruskal decaying into soliton problem. The following schemes were tested on the ten point stencil, using $N = 512$ mesh points except for the high resolution reference run (solid line) for which $N = 2048$ points were used: Non-invariant standard finite differences (crosses), non-invariant momentum conservative (dots), invariant evolution–projection (open circles), invariant evolution–projection momentum conservative (stars), invariant adaptive with monitor function $\rho = \sqrt{1 + 10^4 u_x^2}$ (upward pointing triangles), invariant adaptive momentum-preserving with monitor function $\rho = \sqrt{1 + 10^4 u_x^2}$ (downward pointing triangles), non-invariant adaptive with monitor function $\rho = \sqrt{1 + 10^2 u_{xx}^2}$ (rightward pointing triangles), non-invariant adaptive momentum-preserving with monitor function $\rho = \sqrt{1 + 10^2 u_{xx}^2}$ (leftward pointing triangles). See Table 7.1 for a quantification of these numerical experiments that are visually practically indistinguishable.

7.4.2 Exact algebraic solution

As was discussed in Section 7.3.6, the invariant Lagrangian schemes (7.32) and (7.33) using (7.34) are exact for the Galilean invariant solution (7.10). We verify this by numerically computing this solution and calculating the l_∞-norm and the RMSE. The l_∞-norm is the maximum absolute difference between the numerical solution u_{num} and the exact analytical solution u_{exact} calculated at the discrete mesh points.

The results as seen in Table 7.2 show that we achieve machine precision (i.e. the errors come only from rounding) with the different invariant schemes introduced in Section 7.3 but do not get comparable accuracy with the standard schemes. Table 7.2 also highlights that the evolution–projection method (both invariant and invariant momentum conserving) reproduces the exact solution up to machine precision as well.

Scheme	RMSE
Non-invariant standard	0.0138
Non-invariant standard $\mathcal{M}$-cons	0.0139
Invariant evolution–projection	0.189
Invariant evolution–projection $\mathcal{M}$-cons	0.202
Invariant adaptive ($\rho(\alpha, u_x)$ with $\alpha = 1e4$)	0.0142
Invariant adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_x)$ with $\alpha = 1e4$)	0.0139
Non-invariant adaptive ($\rho(\alpha, u_{xx})$ with $\alpha = 1e2$)	0.0144
Non-invariant adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_{xx})$ with $\alpha = 1e2$)	0.0138

Table 7.1: Numerical errors for the Zabusky–Kruskal problem. The RMSE is based on a high resolution integration using $N = 2048$ mesh points and a time step $\Delta t = 3.125 \cdot 10^{-7}$ in the non-invariant standard numerical scheme for the KdV equation. All other schemes use $N = 512$ mesh points with time step $\Delta t = 5 \cdot 10^{-6}$.

Scheme	l_∞-norm	RMSE
Non-invariant standard	6.76e-6	2.39e-6
Non-invariant standard $\mathcal{M}$-cons	7.77e-6	3.30e-6
Invariant Lagrangian	4.73e-13	2.02e-13
Invariant evolution–projection	2.13e-14	7.93e-15
Invariant Lagrangian $\mathcal{M}$-cons	9.73e-13	4.02e-14
Invariant evolution–projection $\mathcal{M}$-cons	5.15e-14	1.31e-14

Table 7.2: Comparison of errors for the various ten point schemes to reproduce the exact solution (7.10) evaluated at $t = 2$. All schemes use $N = 35$ mesh points on the domain $[0, 20]$ and time steps of $\Delta\tau = 0.001$. The starting time of the integrations is $t_0 = 1$.

This solution, being monotonously increasing, is one of the few where the Lagrangian moving mesh points cause no instability over a longer period of time. No interpolation or adaptation is therefore needed to get an exact solution at any time. We should also stress that for this simple solution the adaptive schemes would coincide with the standard scheme as $u_x = 1/t$ and $u_{xx} = 0$ thus reducing the discretized equidistribution principle (7.38) for both the invariant and non-invariant mesh density functions (7.39) and (7.41) to $x_{i+1}^{n+1} - x_i^{n+1} = x_i^{n+1} - x_{i-1}^{n+1}$.

While integrating such a simple function is trivial, this example shows the compatibility of preserving symmetries and obtaining exact discrete solutions.

7.4.3 Cnoidal wave and soliton solution

For any numerical scheme, one important test is to verify consistency and the order of convergence. To verify the order of the numerical schemes proposed in this paper, we take cnoidal wave periodical solution of the form

$$u = (a - b)cn^2(\omega(x + bt), k) \tag{7.42}$$

where $a = 3.332$, $b = -0.784$, $c = -2.548$, $k = \sqrt{\frac{a-b}{a-c}} = \sqrt{0.7}$ and $\omega = \sqrt{\frac{a-c}{12}} = 0.7$

We then vary the total number of mesh points $n \in \{16, 24, 32, 38\}$ and measure for each associated numerical experiment the error characterized by the l_∞-norm of the difference between the numerical and the discrete analytical solutions. A linear regression of log(n) vs. log(error) gives a slope corresponding to the order of convergence in $\mathcal{O}(n^p)$. All our ten point schemes should theoretically converge as $\mathcal{O}(n^{-2})$ and we notice in Table 7.3 that this is numerically effectively the case.

Scheme	p in $\mathcal{O}(N^p)$
Non-invariant standard	-2.00
Non-invariant standard $\mathcal{M}$-cons	-2.05
Invariant Lagrangian	-2.13
Invariant Lagrangian $\mathcal{M}$-cons	-2.11
Invariant evolution–projection	-1.91
Invariant evolution–projection $\mathcal{M}$-cons	-2.04
Invariant adaptive ($\rho(\alpha, u_x)$ with $\alpha = 5e6$)	-2.05
Invariant adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_x)$ with $\alpha = 5e6$)	-2.05
Non-invariant adaptive ($\rho(\alpha, u_{xx})$ with $\alpha = 1e6$)	-2.00
Non-invariant adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_{xx})$ with $\alpha = 1e6$)	-2.02

Table 7.3: Convergence tests are done for the cnoidal solution over one spatial period at time $t = 0.2$ with time step $\Delta t = 10^{-4}$. All schemes use the ten point lattice. The integrations are done using $N = \{16, 24, 32, 48\}$ points. We confirm that all schemes converge as $\mathcal{O}(N^{-2})$ in the l_∞-norm and are therefore consistent.

To assess not only the order of the numerical schemes but also the absolute approximation errors in Table 7.4 we present the RMSE comparing the numerical solution against the exact cnoidal wave solution as given in (7.42). As a second example, we also compare against the soliton solution

$$u = \frac{3\nu}{\cosh^2\left(\frac{1}{2}\sqrt{\nu}(x - \nu t)\right)} \tag{7.43}$$

with $\nu = 7$. In addition to the approximation error we also monitor the change in momentum $\Delta\mathcal{M}$ over the integration period.

By computing the RMSE of the different invariant and non-invariant schemes, in Table 7.4 we are able to affirm that invariant and non-invariant schemes give roughly the same approximation errors. The invariant adaptive and non-invariant adaptive scheme give comparable accuracy while the standard scheme is slightly better than the Lagrangian scheme. We confirm that the invariant ten point scheme gives better accuracy than the invariant explicit scheme on the five point lattice as expected. The basic projection method using parabolic interpolation helps to reduce the error and allows using longer integration times. Optimizing the adaptation parameter α is possible as well (see [11]) and could lead to error improvements. This will however not be pursued here.

7.4.4 Double soliton solution and Galilean invariance

The above numerical experiments show that in terms of accuracy the invariant and the non-invariant schemes are mostly comparable (except for the exact solution (7.10)). Still, from the

	Cnoidal wave		Soliton	
Scheme	RMSE	$\Delta\mathcal{M}$	RMSE	$\Delta\mathcal{M}$
Non-inv standard	3.98e-3	1.96e-14	9.56e-2	3.20e-14
Non-inv standard $\mathcal{M}$-cons	1.52e-3	2.66e-14	3.38e-2	7.11e-14
Inv 5 point explicit Lagrangian	4.59e-2	8.07e-3	0.439	0.367
Inv Lagrangian	7.69e-3	1.26e-3	0.346	0.166
Inv Lagrangian $\mathcal{M}$-cons	9.91e-3	2.31e-14	0.436	3.91e-14
Inv evolution–projection	4.93e-3	8.19e-4	0.288	7.78e-2
Inv evolution–projection $\mathcal{M}$-cons	5.58e-3	7.59e-4	0.327	4.08e-2
Non-inv adaptive ($\rho(\alpha, u_{xx})$, $\alpha = 1e6$)	3.92e-3	2.71e-6	—	—
Non-inv adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_{xx})$, $\alpha = 1e6$)	1.57e-3	2.31e-14	—	—
Inv adaptive ($\rho(\alpha, u_x)$, $\alpha = 5e6$)	3.99e-3	1.54e-5	—	—
Inv adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_x)$, $\alpha = 5e6$)	1.48e-3	3.38e-14	—	—
Non-inv adaptive ($\rho(\alpha, u_{xx})$, $\alpha = 1e4$)	—	—	9.49e-2	9.12e-4
Non-inv adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_{xx})$, $\alpha = 1e4$)	—	—	2.94e-2	4.97e-14
Inv adaptive ($\rho(\alpha, u_x)$, $\alpha = 1e4$)	—	—	9.28e-2	5.74e-4
Inv adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_x)$, $\alpha = 1e4$)	—	—	0.682	3.55e-14

Table 7.4: Errors of different schemes for the cnoidal wave and soliton solutions. All schemes use the ten point lattice unless otherwise stated. Time steps are always $\Delta t = 10^{-4}$. The cnoidal wave is integrated over one period up to $t = 0.2$ while the soliton is computed up to $t = 0.05$ on the domain $[-4, 4]$. The short integration time is to allow using the purely Lagrangian method. For both integration $N = 48$ total mesh points are used. The projection method is parabolic interpolation. The suitable adaptation parameter depends on both the form of the monitor function and the initial conditions.

physical point of view, the additional advantage of the invariant schemes over the standard ones is the preservation of Galilean invariance. In particular, Galilean invariance in a numerical scheme implies that applying a boost to any solution does not change the discrete numerical solution. Hence, the numerical solutions can be obtained in any constantly moving reference frame. This can be an important property in practical applications, see e.g. [4] and references therein for applications of this property to hydrodynamics.

To numerically verify Galilean invariance in the proposed invariant schemes, we integrate the double soliton solution over a short period of time and apply a boost to the invariant and the non-invariant schemes. The following form of the double soliton solution is used:

$$\tilde{u}(t, \tilde{x}) = 12\frac{\partial^2}{\partial x^2}\ln(1 + B_1 e^{iQ_1} + B_2 e^{iQ_2} + AB_1B_2 e^{i(Q_1+Q_2)}) + c,$$
$$Q_1 = a_1\tilde{x} - a_1^3 t, \quad a_2\tilde{x} - a_2^3 t, \quad A = \left(\frac{a_1 - a_2}{a_1 + a_2}\right)^2$$

where $a_1 = -2i$, $a_2 = -i$, $B_1 = 10000$, $B_2 = 1$, $\{\tilde{u}, \tilde{x} = x - ct\}$ belong to the moving reference frame and c is the speed of the moving reference frame.

Two sets of numerical experiments are carried out for each scheme. One in a resting reference frame, i.e. $c = 0$ and one in a constantly moving reference frame, $c \neq 0$. After the end of each

integration both solutions are compared to each other. Galilean invariance implies that both solutions must coincide up to machine precision.

By increasing the strength of the boost and by computing the RMSE, we observe an increase in the error for the non-invariant momentum-conserving scheme while the invariant adaptive and momentum-conserving scheme is largely unaffected, see Table 7.5 for quantification. The Galilean boosted solution for the invariant scheme in Fig. 7.3 is identical to its equivalent in the resting reference frame and visually confirms the Galilean invariance of this scheme, a major physical property lost when using standard non-invariant schemes. For the other invariant and non-invariant schemes the results are essentially the same and are hence not presented here.

	RMSE compared to c=0 solution	
$c/\Delta x$	Non-inv standard $\mathcal{M}$-cons	Inv adaptive $\mathcal{M}$-cons ($\rho(\alpha, u_x)$, $\alpha = 1e4$)
-10	2.14e-1	1.82e-12
-1	2.29e-2	1.07e-12
0	0	0
1	2.19e-2	8.76e-13
5	0.103	3.15e-12
10	0.202	1.86e-12
30	0.564	1.03e-12

Table 7.5: RMSE comparing the resting reference solution ($c = 0$) to a constantly moving solution ($c \neq 0$) for the non-invariant momentum-preserving scheme (left) and the invariant adaptive momentum-preserving scheme (right). Integrations were done up to $t = 1$ using the time step $\Delta t = 10^{-3}$ and $N = 128$ points. It can be seen that varying the speed c of the reference frame leads to significantly different solutions for the standard scheme as measured through the RMSE while for the invariant scheme the RMSE stays approximately constant and is due to rounding only.

7.5 Conclusions

In this paper we have constructed invariant numerical schemes for the Korteweg–de Vries equation. While some invariant numerical schemes have already been constructed for this equation in the past [7, 14], to the best of our knowledge this is the first time that actual numerical experiments have been carried out for the KdV equation using such schemes. We found that these existing schemes, all Lagrangian in nature, can develop tangling meshes and hence may not allow integration beyond some fixed time limit. A remedy for these schemes is provided through invoking them in an evolution–projection framework. As shown for several test cases, these evolution–projection schemes can produce numerical solutions for the KdV equation without being restricted by the development of mesh problems.

In addition, we have proposed several Eulerian numerical schemes most notably by using ideas of adaptive moving mesh methods. These schemes are attractive in that they link the required moving meshes (to preserve Galilean invariance) to the development of pronounced features of the numerical solution. Hence, such schemes are capable of tracking developing shots, blow-ups

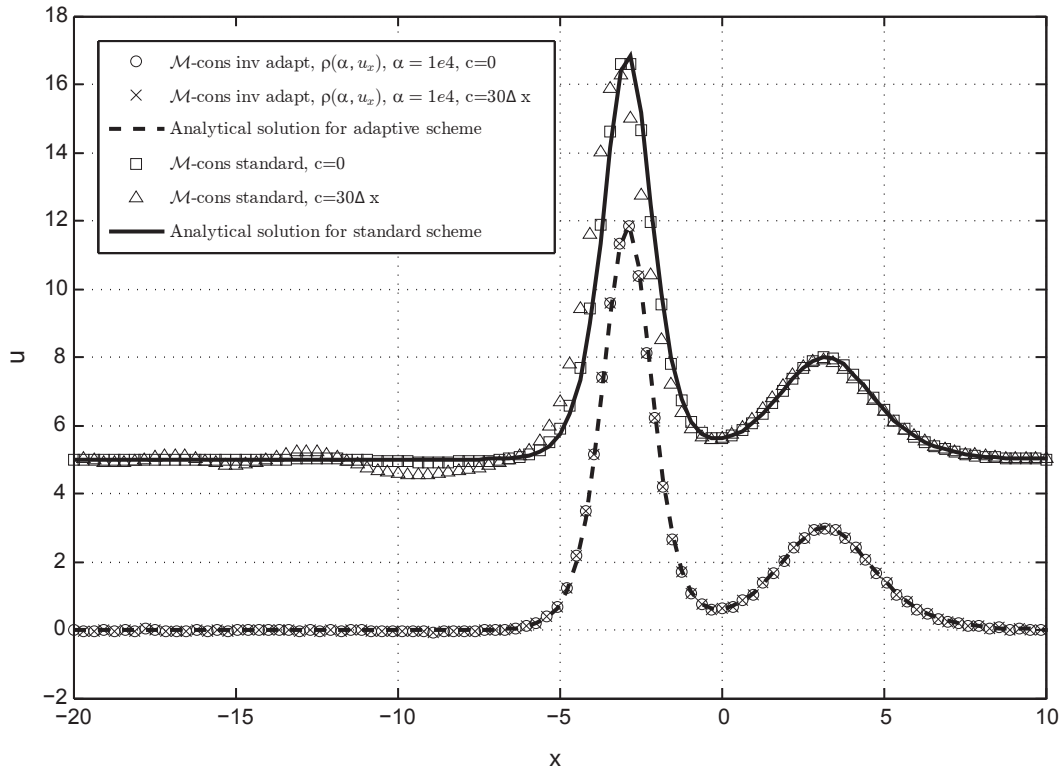

Figure 7.3: Double soliton solution at time $t = 0.1$ computed using the non-invariant and invariant adaptive momentum-conservative schemes. Time step used is $\Delta t = 10^{-4}$ with $N = 128$ grid points. The non-invariant solutions are shifted with respect to the invariant solutions for the sake of comparison. While the non-invariant scheme in a resting reference frame (open squares) approximates closely the exact solution (dashed line), using this scheme in a constantly moving reference frame (triangles) leads to large deviations from the true solution. For the invariant scheme, both the solution in the resting reference frame (open circles) and in a constantly moving reference frame (crosses) are in good accordance with the exact solution (solid line).

etc. Furthermore, we have shown that it is possible to develop invariant numerical schemes that also preserve momentum.

The results of the present article can be viewed from two points of view, physics and numerical analysis.

From the point of view of a physicist it is obvious that symmetries of a physical system and of the equations describing it are important. We have confirmed that for the KdV equation it is possible to write a discrete system that in addition to translations and dilations preserves Galilei invariance. The method provides a good numerical description of the exact solutions considered. Moreover, as analyzed in Section 4.4, a solution calculated in the rest frame and then boosted to a moving frame will be numerically the same as one calculated in the moving frame. In a non invariant scheme the two solutions will be different and that is not acceptable physically.

From the point of view of numerical analysis this article fits into the field of Geometric Integration [26, 29, 39] the aim of which is to incorporate qualitative features of a specific equation into its discretization and numerical solution. We concentrate on the preservation of point symmetries. For the KdV equation we have found that in terms of accuracy the invariant

and non invariant schemes give comparable results.This is in striking contrast to symmetry-preserving integrators for ODES, where invariant schemes can perform significantly better than their non-invariant counterparts, especially for solutions with singularities [6, 7, 48]. For the KdV equation, one possible explanation for this discrepancy is that the maximal Lie invariance group is of rather simple structure, with three of the four admitted one-parameter symmetry transformations (shifts in space and time as well as dilations) already preserved by standard numerical schemes for this equation. Hence, the only difference between the non-invariant and invariant schemes for the KdV equation is whether Galilean invariance is admitted or not. Another point is that probably the most important feature of the KdV equation is its integrability [1] which is not conserved in our discretization.

For other equations point symmetries can be much richer. For example the Liouville equation has an infinite dimensional Lie point symmetry group. In a recent study [34] it was shown that a discretization preserving invariance under the maximal finite dimensional subgroup of the symmetry group gives much better numerical results than other discretizations.

We hope to carry out a more numerically focused study of invariant numerical schemes in the near future and to tackle more challenging problems such as the use of non-smooth data as suggested by Chen and Olver in connection with nonlinear dispersive quantization [12, 13].

Acknowledgements

This research was supported by the Austrian Science Fund (FWF), project J3182–N13 (AB). AB is a recipient of an APART Fellowship of the Austrian Academy of Sciences. The research of PW was partially supported by NSERC of Canada. He thanks the European Union Research Executive Agency for the award of a Marie Curie International Incoming Research Fellowship making his stay at Università Roma Tre possible. He thanks the Department of Mathematics and Physics of the Università Roma Tre for hospitality.

Bibliography

[1] Ablowitz M.J. and Clarkson P.A., *Solitons, Nonlinear Evolution Equations and Inverse Scattering*, vol. 149, Cambridge University Press, Cambridge, 1991.

[2] Bakirova M.I., Dorodnitsyn V.A. and Kozlov R.V., Symmetry-preserving discrete schemes for some heat transfer equations, *J. Phys. A* **30** (1997), 8139–8155.

[3] Bihlo A. and Nave J.C., Invariant discretization schemes using evolution–projection techniques, *SIGMA* **9** (2013), 23 pages, arXiv:1209.5028.

[4] Bihlo A. and Nave J.C., Convecting reference frames and invariant numerical models, *J. Comput. Phys.* **272** (2014), 656–663, arXiv:1301.5955.

[5] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[6] Bourlioux A., Cyr-Gagnon C. and Winternitz P., Difference schemes with point symmetries and their numerical tests, *J. Phys. A* **39** (2006), 6877–6896.

[7] Bourlioux A., Rebelo R. and Winternitz P., Symmetry preserving discretization of SL (2, R) invariant equations, *J. Nonlinear Math. Phys* **15** (2008), 362–72.

[8] Budd C. and Dorodnitsyn V.A., Symmetry-adapted moving mesh schemes for the nonlinear Schrödinger equation, *J. Phys. A* **34** (2001), 10387–10400.

[9] Budd C.J. and Collins G.J., Symmetry based numerical methods for partial differential equations, in *Numerical analysis 1997 (Dundee)*, vol. 380 of *Pitman Res. Notes Math. Ser.*, Longman, Harlow, pp. 16–36, 1998.

[10] Budd C.J. and Iserles A., Geometric integration: numerical solution of differential equations on manifolds, *Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* **357** (1999), 945–956.

[11] Byrd P.F. and Friedman M.D., *Handbook of Elliptic Integrals for Engineers and Scientists*, Springer, Berlin, 1971.

[12] Chen G. and Olver P.J., Dispersion of discontinuous periodic waves, *Proc. Roy. Soc. Ser. A* **469** (2013), 20120407.

[13] Chen G. and Olver P.J., Numerical simulation of nonlinear dispersive quantization, *Discrete Cont. Dyn. Syst. A* **34** (2013), 991–1008.

[14] Dorodnitsyn V., *Invariant discrete model for the Korteweg–de Vries equation*, *Tech. Rep. CRM-2187*, Centre de recherches mathématiques, Université de Montréal, 1994.

[15] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[16] Dorodnitsyn V., Kaptsov E., Kozlov R. and Winternitz P., First integrals of ordinary difference equations beyond Lagrangian methods, ArXiv:1311.1597, 2013.

[17] Dorodnitsyn V., Kozlov R. and Winternitz P., Lie group classification of second-order ordinary difference equations, *J. Math. Phys.* **41** (2000), 480–504.

[18] Dorodnitsyn V., Kozlov R. and Winternitz P., Continuous symmetries of Lagrangians and exact solutions of discrete equations, *J. Math. Phys.* **45** (2004), 336–359.

[19] Dorodnitsyn V.A., Transformation groups in mesh spaces, *J. Sov. Math.* **55** (1991), 1490–1517.

[20] Dorodnitsyn V.A. and Kozlov R., A heat transfer with a source: the complete set of invariant difference schemes, *J. Nonlin. Math. Phys.* **10** (2003), 16–50.

[21] Dorodnitsyn V.A. and Winternitz P., Lie point symmetry preserving discretizations for variable coefficient Korteweg–de Vries equations, *Nonlin. Dyn.* **22** (2000), 49–59.

[22] Dubrovin B.A., Matveev V.B. and Novikov S.P., Non-linear equations of Korteweg-de Vries type, finite-zone linear operators, and Abelian varieties, *Russ. Math. Surveys* **31** (1976), 59.

[23] Dubrovin B.A. and Novikov S.P., Periodic and conditionally periodic analogs of the many-soliton solutions of the Korteweg-de Vries equation, *Sov. Phys. JETP* **40** (1975), 1058–1063.

[24] Gardner C.S., Greene J.M., Kruskal M.D. and Miura R.M., Method for solving the Korteweg–de Vries equation, *Phys. Rev. Lett.* **19** (1967), 1095–1097.

[25] Gromak V.I., Laine I. and Shimomura S., *Painlevé Differential Equations in the Complex Plane*, vol. 28 of *de Gruyter Studies in Mathematics*, Walter de Gruyter, Berlin, 2002.

[26] Hairer E., Lubich C. and Wanner G., *Geometric Numerical Integration: Structure-preserving Algorithms for Ordinary Differential Equations*, Springer, Berlin, 2006.

[27] Huang W. and Russell R.D., *Adaptive Moving Mesh Methods*, Springer, New York, 2010.

[28] Ince E.L., *Ordinary Differential Equations*, Dover Publications, Mineola, NY.

[29] Iserles A., *A first course in the numerical analysis of differential equations*, 44, Cambridge University Press, 2009.

[30] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Phys. D* **237** (2008), 243–254.

[31] Kim P. and Olver P.J., Geometric integration via multi-space, *Regul. Chaotic Dyn.* **9** (2004), 213–226.

[32] Krichever I.M. and Novikov S.P., Holomorphic bundles over algebraic curves and non-linear equations, *Russ. Math. Surveys* **35** (1980), 53–79.

[33] Leimkuhler B. and Reich S., *Simulating Hamiltonian Dynamics*, Cambridge University Press, Cambridge, 2004.

[34] Levi D., Martina L. and Winternitz P., Lie-point symmetries of the discrete Liouville equation, ArXiv:1407.4043, 2014.

[35] Levi D., Tremblay S. and Winternitz P., Lie symmetries of multidimensional difference equations, *J. Phys. A* **34** (2001), 9507–9524.

[36] Levi D. and Winternitz P., Lie groups and numerical solutions of differential equations: Invariant discretization versus differential approximation, *Acta Polytechnica* **53**, 438–443.

[37] Levi D. and Winternitz P., Continuous symmetries of discrete equations, *Phys. Lett. A* **152** (1991), 335–338.

[38] Levi D. and Winternitz P., Continuous symmetries of difference equations, *J. Phys. A* **39** (2006), R1–R63.

[39] Marsden J.E. and West M., Discrete mechanics and variational integrators, *Acta Numer.* **10** (2001), 357–514.

[40] Nave J.C., Rosales R.R. and Seibold B., A gradient-augmented level set method with an optimally local, coherent advection scheme, *J. Comput. Phys.* **229** (2010), 3802–3827.

[41] Okubo S. and Marshak R.E., Velocity dependence of the two-nucleon interaction, *Ann. Physics* **4** (1958), 166–179.

[42] Olver P.J., *Application of Lie groups to Differential Equations*, Springer, New York, 2000.

[43] Olver P.J., Geometric foundations of numerical algorithms and symmetry, *Appl. Algebra Engrg. Comm. Comput.* **11** (2001), 417–436.

[44] Polyanin A.D. and Zaitsev V.F., *Handbook of Nonlinear Partial Differential Equations*, Chapman & Hall/CRC, New York, 2004.

[45] Puzikov L., Ryndin R. and Smorodinsky J., Construction of the scattering matrix of a two-nucleon system, *Nuclear Physics* **3** (1957), 436–445.

[46] Rebelo R. and Valiquette F., Symmetry preserving numerical schemes for partial differential equations and their numerical tests, *J. Difference Equ. Appl.* **19** (2013), 738–757.

[47] Rebelo R. and Valiquette F., Invariant discretization of partial differential equations admitting infinite-dimensional symmetry groups, ArXiv:1401.4380, 2014.

[48] Rebelo R. and Winternitz P., Invariant difference schemes and their application to invariant ordinary differential equations, *J. Phys. A* **42** (2009), 454016 (10 pages).

[49] Rodriguez M.A. and Winternitz P., Lie symmetries and exact solutions of first-order difference schemes, *J. Phys. A* **37** (2004), 6129.

[50] Seibold B., Rosales R.R. and Nave J.C., Jet schemes for advection problems, *Discrete Contin. Dyn. Syst. Ser. B* **17** (2012), 1229–1259.

[51] Shokin Y., *The Method of Differential Approximation*, Springer, Berlin, 1983.

[52] Valiquette F. and Winternitz P., Discretization of partial differential equations preserving their physical symmetries, *J. Phys. A* **38** (2005), 9765–9783.

[53] Winternitz P., Symmetries of discrete systems, in *Discrete Integrable Systems*, vol. 644 of *Lecture Notes in Physics*, edited by B. Grammaticos, T. Tamizhmani and Y. Kosmann-Schwarzbach, Springer Berlin Heidelberg, pp. 185–243, 2004.

[54] Winternitz P., Dorodnitsyn V., Kaptsov E. and Kozlov R., First integrals of ordinary difference equations which do not possess a variational formulation, *Dokl. Math.* **89** (2014), 106–109, arXiv:1307.7585.

[55] Yanenko N.N. and Shokin Y.I., Correctness of first differential approximations of difference schemes, *Doklady Akademii Nauk SSSR* **182** (1968), 776.

[56] Zabusky N.J. and Kruskal M.D., Interaction of solitons in a collisionless plasma and the recurrence of initial states, *Phys. Rev. Lett* **15** (1965), 240–243.

Chapter 8

Invariant meshless discretization schemes

Alexander Bihlo[†‡]

[†] *Centre de recherches mathématiques, Université de Montréal, C.P. 6128, succ. Centre-ville, Montréal (QC) H3C 3J7, Canada*

[‡] *Department of Mathematics and Statistics, McGill University, 805 Sherbrooke W., Montréal (QC) H3A 2K6, Canada*

E-mail*: bihlo@crm.umontreal.ca*

A method is introduced for the construction of meshless discretization schemes which preserve Lie symmetries of the differential equations that these schemes approximate. The method exploits the fact that equivariant moving frames provide a way of associating invariant functions to non-invariant functions. An invariant meshless approximation of a nonlinear diffusion equation is constructed. Comparative numerical tests with a non-invariant meshless scheme are presented. These tests yield that invariant meshless schemes can lead to substantially improved numerical solutions compared to numerical solutions generated by non-invariant meshless schemes.

8.1 Introduction

Invariant discretization schemes have received increasing attention over the past 20 years, see e.g. [6, 7, 13, 17]. Such schemes are attractive in that they preserve an important property of a system of differential equations, namely its maximal Lie invariance group G or at least a certain physically interesting subgroup of G.

One of the most distinct properties of invariant discretization schemes for evolution equations is that they in general require the usage of a moving discretization mesh. The necessity of using non-orthogonal and/or non-stationary meshes considerably complicates the construction and analysis of invariant numerical schemes, especially in the multi-dimensional case. Special techniques can be used to overcome this problem, such as the symmetry-preserving discretization in computational coordinates [4, 11] or invariant interpolation schemes [3]. A problem with the

former technique is that in the higher-dimensional case quite some computational overhead might be required to construct a proper mapping from the computational domain to the physical space of the system of differential equations.

On the other hand, moving meshes or grids that are adapted to complicated domain geometries which hamper the straightforward use of finite differences or related discretization strategies are not new in the numerical analysis of differential equations. In fact, constructing, storing and modifying discretization grids is costly and hence is attributed as one of the main drawbacks of the otherwise popular finite element method [5, 15, 16]. Indeed, often a large fraction of the computational time spent for the numerical integration of a system of differential equations is consumed by the construction of the discretization mesh itself.

This is why a new class of discretization schemes is steadily growing in importance over the past years, namely so-called *meshless schemes*. The main observations on which meshless methods rest is that no information on the connectivity of the single nodes at which the numerical solution is sought is required in order to discretize a system of differential equations [5, 15, 16]. *All the information needed to approximate the various derivatives is already included in the nodes themselves.* This makes meshless methods attractive to (at least partially) avoid the computational overhead required by the construction of discretization meshes.

It is then obvious to ask whether meshless methods could be employed in the construction of invariant numerical discretization schemes as well. This would allow one to (partially) bypass the complication one faces in attempting to find discrete approximations of a system of differential equations that preserve the symmetries of that system.

It is the purpose of this paper to describe an algorithm for the construction of *invariant meshless discretization schemes*. The key idea on which our construction relies is a property of *equivariant moving frames* to send a given function to an invariant function [9, 10]. This property was successfully exploited to construct invariant finite difference schemes for partial differential equations starting from non-invariant schemes [3, 12, 21]. In the present paper we extend this method to meshless discretization schemes.

As what concerns the organization of this paper, in Section 8.2 we describe the general procedure for finding invariant meshless discretization schemes. This method is applied in Section 8.3 to construct an invariant meshless scheme for a nonlinear diffusion equation. Numerical tests comparing the invariant with the non-invariant meshless scheme are carried out in Section 8.4. The conclusion and some thoughts for further research directions are presented in Section 8.5.

8.2 Invariant meshless discretization schemes

There is not a unique way of constructing meshless approximations to differential equations. In fact, a number of different strategies to discretize a differential equation without or only partial usage of a discretization lattice are used, such as meshless (generalized) finite differences, smooth particle hydrodynamics, the element free Galerkin method or the meshless local Petrov–Galerkin method. For a review of these and further techniques, see e.g. [2, 5, 14–16] and references therein.

In the present paper we exclusively work with meshless discretizations based on *meshless finite differences*. We stress, though, that similar techniques as introduced below could be applied to other meshless methods that discretize a system of differential equations in the strong form.

Meshless finite differences basically rest on the expansion of a function $u\colon \mathbb{R}^p \to \mathbb{R}$ in a multi-dimensional Taylor series around the node x^0,

$$u(x) = \sum_\alpha \frac{1}{\alpha!} u_\alpha|_{x^0} (x - x^0)^\alpha,$$

where $x = (x_1, \ldots, x_p)$ and $\alpha = (\alpha_1, \ldots, \alpha_p)$ is a multi-index, $\alpha_j \in \mathbb{N}_0$, $u_\alpha = \partial^{|\alpha|} u / \partial x_1^{\alpha_1} \cdots \partial x_p^{\alpha_p}$, $|\alpha| = \alpha_1 + \cdots + \alpha_p$, $\alpha! = \alpha_1! \cdots \alpha_p!$ and $(x - x^0)^\alpha = (x_1 - x_1^0)^{\alpha_1} \cdots (x_p - x_p^0)^{\alpha_p}$.

Truncating this series to the mth order derivatives one is left with an expansion that includes $s = (p+m)!/(p!m!)$ coefficients. Thus, in theory s nodes x^j are needed to solve for the s derivatives u_α evaluated at the node x^0 from the linear system

$$u(x^j) = u^j = \sum_\alpha \frac{1}{\alpha!} u_\alpha|_{x^0} (x^j - x^0)^\alpha, \tag{8.1}$$

$j = 1, \ldots, s$. The s nodal points (x^j, u^j) are usually chosen to be neighboring points lying in the ball of radius r centered around the node x^0. In particular, the nodes are *not* required to lie on a predefined, topologically connected mesh which makes the method truly meshless.

The practical problem that can arise in this construction is that for certain distributions of the s nodes (e.g. all points lying on a line), the system (8.1) cannot be solved for the required derivatives $u_\alpha|_{x^0}$. A possible ad hoc remedy for this problem is to include more than s points in the system (8.1), i.e. to over-determine it [5, 14]. The derivatives $u_\alpha|_{x^0}$ then follow from the least squares solution of (8.1), which reads

$$(u^{(m)})^{\mathrm{d}}|_{x^0} = (S^{\mathrm{T}} W S)^{-1} S^{\mathrm{T}} W b, \tag{8.2}$$

where

$$(u^{(m)})^{\mathrm{d}}|_{x^0} = \begin{pmatrix} u^0 \\ u^{\mathrm{d}}_{x_1} \\ \vdots \\ u^{\mathrm{d}}_\alpha \end{pmatrix}, \quad S = \begin{pmatrix} 1 & \Delta x_1^1 & \cdots & \Delta x_\alpha^1 \\ 1 & \Delta x_1^2 & \cdots & \Delta x_\alpha^2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \Delta x_1^k & \cdots & \Delta x_\alpha^k \end{pmatrix}, \quad b = \begin{pmatrix} u^1 \\ u^2 \\ \vdots \\ u^k \end{pmatrix}.$$

The vector $(u^{(m)})^{\mathrm{d}}|_{x^0}$ contains the s derivatives of the truncated Taylor series evaluated at the node x^0 where u^{d}_α is the highest derivative occurring. The matrix S is build from the associated coefficients of these derivatives where $\Delta x_\alpha^j = (x^j - x^0)^\alpha$. The vector b includes the $k \geq s$ functions values u^j of the nodes in the ball of radius r centered at x^0. A (diagonal) weight matrix W is included in the least squares solution for the geometrical reason to give greater weight to the points x^j closer to x^0. More details on this construction can be found in [5, 14]. The extension to vector-valued functions $u = (u_1, \ldots, u_q)$ is straightforward.

The meshless approximated derivatives u^{d}_α can be used to discretize a system of differential equations $\mathcal{L}\colon \Delta_l(x, u^{(m)}) = 0$, $l = 1, \ldots, L$, where $u^{(m)}$ includes all the derivatives of u with respect to x up to order m as well as u itself. This leads to a meshless numerical scheme $\mathcal{D}\colon D_l(x^j, (u^{(m)})^{\mathrm{d}}) = 0$, where $(u^{(m)})^{\mathrm{d}}$ denote the discretizations of derivatives in $u^{(m)}$ using Eq. (8.2). If the system $\mathcal{L}$ includes derivatives of u with respect to the time t then $\mathcal{D}$ in addition to the meshless spatial derivatives requires discretizations of the time-dependent derivatives.

We now briefly describe the method of *invariantization* using *moving frames*. An extended discussion can be found in several excellent papers on that subject, including [9, 10, 17, 18, 21].

Definition 8.1. A (right) *moving frame* ρ is a smooth map $\rho\colon M \to G$ from a manifold M to a finite dimensional Lie group G acting on M with the property that

$$\rho(g \cdot z) = \rho(z)g^{-1}, \tag{8.3}$$

for all $z \in M$ and $g \in G$.

The theorem on moving frame requires a group action to be free and regular in order to guarantee the existence of the moving frame ρ. Although the group G in Definition 8.1 is restricted to be finite dimensional, this restriction is in fact only apparent as the theory of moving frames is already formulated for infinite dimensional Lie (pseudo)groups, see e.g. [19].

If the group action of G is not free, it can be made free by constructing the moving frame on a jet space $\mathrm{J}^m = \mathrm{J}^m(M,p)$ of appropriate order m. An alternative way of making a group action free is to extend it to the product action on the subset $M^\diamond$ of the Cartesian product of copies of the original space M, which is defined as $M^\diamond = \{(z^1,\dots,z^k) |\, x^i \neq x^j \text{ for all } i \neq j\}$, where $z^j = (x^j, u^j)$ are the nodal points. The joint product action is simply the component-wise action, $g \cdot (z^1,\dots,z^k) = (g \cdot z^1,\dots, g \cdot z^k)$, see also [3, 17, 21].

Moving frames are determined from a method referred to as *normalization*. For a r dimensional group action, in this procedure one sets up a system of r equations involving the coordinate functions z, where $z \in \{x_i, u^{(m)}\}$ in the case G acts on the mth order jet space J^m or $z \in \{x^j_i, u^j\}$, in case the action of G on $M^\diamond$ is considered. The first possibility leads to a moving frame $\rho^{(m)}$ on the jet space J^m, $\rho^{(m)}\colon \mathrm{J}^m \to G$ while the second possibility leads to a product frame $\rho^\diamond$ on the space $M^\diamond$, $\rho^\diamond\colon M^\diamond \to G$.

The system of r normalization conditions is chosen in such a manner as to determine a submanifold of J^m (or $M^\diamond$) which intersects the group orbits only once and transversally. One usually sets r of the coordinate functions z to appropriately chosen constants, i.e. the normalization equations are $z_1 = c_1,\dots,z_r = c_r$, although in the discrete case it is beneficial to set combinations of the coordinate functions to constants, see the example in Section 8.3. Then one replaces these equations with their respective transformed forms, i.e. $Z_1 = g\cdot z_1 = c_1,\dots,Z_r = g\cdot z_r = c_r$, and solves this algebraic system for the group parameters $g = (\varepsilon_1,\dots,\varepsilon_r)$ in terms of z. The result of this construction is the right moving frame $\rho^{(m)}$ (or $\rho^\diamond$).

For the present purpose the most important property of moving frames is that they define a map from a given (non-invariant) function to a G-invariant function.

Definition 8.2. The *invariantization* of a real-valued function $f\colon M \to \mathbb{R}$ using the (right) moving frame ρ is the function $\iota(f)$, which is defined as $\iota(f)(z) = f(\rho(z)\cdot z)$.

Invariantization is the key for constructing invariant discretization schemes. To accomplish this, the product frame $\rho^\diamond$ is computed using the symmetry group G of the system of differential equations $\mathcal{L}$ and extending its action to the joint space $M^\diamond$. The product frame should be compatible with the moving frame $\rho^{(m)}$, which is determined for the action of G prolonged to the jet space J^m. Compatibility means that $\rho^\diamond \to \rho^{(m)}$ in the continuous limit $x^j \to x$. This is achieved by computing the product frame $\rho^\diamond$ using the discretized form of the normalization conditions that are used to construct the moving frame $\rho^{(m)}$, see [3, 17, 21] for more details.

With the moving frame $\rho^\diamond$ at hand one can invariantize any standard numerical scheme $\mathcal{D}$ which approximates a system of differential equations $\mathcal{L}$. The invariant scheme associated with $\mathcal{D}$ is $\iota(\mathcal{D})\colon D_l(\iota(x^j), \iota((u^{(m)})^{\mathrm{d}})) = 0$, $l = 1,\dots,L$, which in the continuous limit

$x^j \to x$ yields the system of differential equations $\mathcal{L}$ expressed in terms of differential invariants, $\iota(\mathcal{L})\colon \Delta_l(\iota(x), \iota(u^{(m)})) = 0$. See [3, 12, 17, 21] for further details.

The extension of the invariantization procedure to meshless discretization schemes is now straightforward. Once the moving frame $\rho^\diamond$ is determined on the space of nodes (x^j, u^j) it can be applied to the least squares solution (8.2) by invariantizing the vectors $(u^{(m)})^{\mathrm{d}}|_{x^0}$ and b and the matrix S in the following way,

$$\iota((u^{(m)})^{\mathrm{d}}|_{x^0}) = \begin{pmatrix} \iota(u^0) \\ \iota(u^{\mathrm{d}}_{x_1}) \\ \vdots \\ \iota(u^{\mathrm{d}}_{\alpha}) \end{pmatrix}, \quad \iota(S) = \begin{pmatrix} 1 & \iota(\Delta x^1_1) & \cdots & \iota(\Delta x^1_\alpha) \\ 1 & \iota(\Delta x^2_1) & \cdots & \iota(\Delta x^2_\alpha) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \iota(\Delta x^k_1) & \cdots & \iota(\Delta x^k_\alpha) \end{pmatrix}, \quad \iota(b) = \begin{pmatrix} \iota(u^1) \\ \iota(u^2) \\ \vdots \\ \iota(u^k) \end{pmatrix}.$$

The invariantized derivatives in $\iota((u^{(m)})^{\mathrm{d}}|_{x^0})$ and the invariantized nodal points $(\iota(x^j), \iota(u^j))$ are sufficient to invariantize any meshless approximation $\mathcal{D}$ of the system $\mathcal{L}$. This is done by replacing the occurring nodal points and meshless discrete derivatives in $\mathcal{D}$ by their respective invariantizations, just in the same way as this is done in the case of finite difference schemes.

An example for this construction is presented in the subsequent section.

8.3 Invariant meshless scheme for a nonlinear diffusion equation

In this section an invariant meshless scheme is constructed for the nonlinear diffusion equation

$$u_t = (u^{-4/3}u_x)_x. \tag{8.4}$$

The meshless Euler forward scheme for Eq. (8.4) is given by

$$\frac{\hat{u} - u}{\Delta t} = -\frac{4}{3}u^{-7/3}(u^{\mathrm{d}}_x)^2 + u^{-4/3}u^{\mathrm{d}}_{xx}, \tag{8.5a}$$

where $\hat{u}$ stands for the value of $u = u^0$ at the subsequent time level of the integration and $\Delta t = \hat{t} - t$ is the (constant) time step. The meshless derivatives u^{d}_x and u^{d}_{xx} of u^0 are evaluated at t and $x = x^0$. Likewise, the meshless leapfrog scheme for Eq. (8.4) reads

$$\frac{\hat{u} - \check{u}}{2\Delta t} = -\frac{4}{3}u^{-7/3}(u^{\mathrm{d}}_x)^2 + u^{-4/3}(u^{\mathrm{d}}_{xx}), \tag{8.5b}$$

where $\check{u}$ is the value of u at the previous time level. In the numerical results reported in Section 8.4 we use the leapfrog scheme (8.5b) and every 20 steps the Euler scheme (8.5a) to suppress the computational mode of the leapfrog integration.

In both these schemes the derivatives u^{d}_x and u^{d}_{xx} are the meshless approximations of the derivatives u_x and u_{xx}. These derivatives are found from the least squares solution (8.2), where in the present one-dimensional case

$$(u^{(4)})^{\mathrm{d}}|_x = \begin{pmatrix} u \\ u^{\mathrm{d}}_x \\ u^{\mathrm{d}}_{xx} \\ u^{\mathrm{d}}_{xxx} \\ u^{\mathrm{d}}_{xxxx} \end{pmatrix}, \quad S = \begin{pmatrix} 1 & \Delta x^1 & \frac{1}{2}(\Delta x^1)^2 & \cdots & \frac{1}{24}(\Delta x^1)^4 \\ 1 & \Delta x^2 & \frac{1}{2}(\Delta x^2)^2 & \cdots & \frac{1}{24}(\Delta x^2)^4 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \Delta x^k & \frac{1}{2}(\Delta x^k)^2 & \cdots & \frac{1}{24}(\Delta x^k)^4 \end{pmatrix}, \quad b = \begin{pmatrix} u^1 \\ u^2 \\ \vdots \\ u^k \end{pmatrix},$$

As a weight matrix we use $W = \mathrm{diag}(\exp(-\mu(\Delta x^j)^2/r^2))$, $j = 1, \ldots, k$ and $\mu = \mathrm{const}$.

The reason for also including the third and fourth derivatives in the vector $(u^{(4)})^{\mathrm{d}}|_x$ is that it increases the order of approximation of the derivatives u^{d}_x and u^{d}_{xx}. If the matrix S is square, it can be directly inferred from the truncated Taylor series that (at least on a uniform grid) including u^{d}_{xxx} and u^{d}_{xxxx} leads to first derivatives with fourth order accuracy and to second derivatives with third order accuracy. Moreover, in [5] it was shown (again for a uniform grid) that the least square solution (S non-square) invoked for finding $(u^{(m)})^{\mathrm{d}}|_x$ does not degrade the accuracy of the approximation.

We now discuss the invariant meshless approximation of Eq. (8.4). The diffusion equation (8.4) admits a five dimensional maximal Lie invariance algebra $\mathfrak{g}$ which is generated by

$$\partial_t, \quad \partial_x, \quad 2t\partial_t + x\partial_x, \quad 2x\partial_x - 3u\partial_u, \quad x^2\partial_x - 3xu\partial_u, \tag{8.6}$$

see [6, 8] where an invariant finite difference discretization for (8.4) was constructed.

Remark 8.1. It was shown in [6, 8] that the maximal Lie invariance group G of Eq. (8.4) associated with $\mathfrak{g}$ cannot be preserved on a uniform discretization mesh. As was indicated in the introduction, the violation of a symmetry (sub)group by a uniform and/or time-space orthogonal grid is rather common in classical finite difference schemes for evolution equations. This is the main reason why it is necessary to develop suitable invariant mesh generation and adaption strategies for such problems and to use discretizations that can operate on practically all distributions of nodes, such as meshless schemes. As a uniform mesh is not consistent with the symmetry group G of Eq. (8.4) we regard the choice of this equation as reasonable to demonstrate the construction of invariant meshless schemes in the simplest possible setting, despite meshless schemes are traditionally constructed mostly for problems in spatial dimension greater than one.

The single vector fields in (8.6) spanning $\mathfrak{g}$ can be exponentiated to one-parameter Lie groups, which can be composed to yield transformations from the five-parameter maximal Lie invariance group G of (8.4). The transformations of G acting on $M = \{(t, x, u)\}$ are of the form

$$T = e^{2\varepsilon_3}(t + \varepsilon_1), \quad X = e^{\varepsilon_3+2\varepsilon_4}\left(\frac{x}{1-\varepsilon_5 x} + \varepsilon_2\right), \quad U = e^{-3\varepsilon_4}(1-\varepsilon_5 x)^3 u. \tag{8.7}$$

The action of G becomes free on the first jet space $\mathrm{J}^1 = \mathrm{J}^1(M, 2)$. The moving frame $\rho^{(1)}$ on J^1 is constructed from the normalization conditions $t = 0$, $x = 0$, $u = 1$, $u_t = 1$ and $u_x = 0$. This allows one to solve for the group parameters $\varepsilon_1, \dots, \varepsilon_5$. The resulting moving frame $\rho^{(1)}$ is

$$\begin{aligned} &\varepsilon_1 = -t, \quad \varepsilon_2 = -\frac{x^2u_x + 3xu}{3u}, \quad \varepsilon_3 = \frac{1}{2}\ln\left(\frac{u_t}{u}\right), \\ &\varepsilon_4 = \ln\left(\frac{u_x^{4/3}}{xu_x + 3u}\right), \quad \varepsilon_5 = \frac{u_x}{xu_x + 3u}. \end{aligned} \tag{8.8}$$

We now turn to the construction of the compatible product frame $\rho^\diamond$. The joint product action of G on $M^\diamond$ follows from evaluating (8.7) at the single nodes (t^j, x^j, u^j), i.e.

$$T^j = e^{\varepsilon_3}(t^j + \varepsilon_1), \quad X^j = e^{\varepsilon_3+2\varepsilon_4}\left(\frac{x^j}{1-\varepsilon_5 x^j} + \varepsilon_2\right), \quad U^j = e^{-3\varepsilon_4}(1-\varepsilon_5 x^j)^3 u^j, \tag{8.9}$$

and similar on the subsequent and previous time layers. It is readily verified that the joint product action leaves invariant the condition for the nodes to remain fixed during the integration,

which is $\hat{x}^j - x^j = 0$. For this reason, fixed nodes do not break the invariance of Eq. (8.4) and we can assume that $\hat{x}^j = x^j$, which we do for the sake of simplicity. The same is true for the equation $t^{j+1} - t^j = 0$, which defines the flatness of the time layers. Hence, $t^{j+1} = t^j = t$.

Before constructing $\rho^\diamond$ it is worthwhile pointing out that the scheme (8.5) is already invariant under the action of the one-parameter groups associated with $\varepsilon_1, \dots, \varepsilon_4$. This can be seen as both the variables t and x^j only arise in the differences Δt and Δx^j, which are obviously invariant under translations in t and x. At the same time, the scaling properties of the scheme (8.5) are the same as of (8.4) and therefore the scale invariance is preserved by scheme (8.5). Thus, the only symmetry transformation which is violated by (8.5) is associated with the group parameter ε_5, i.e. inversions in x. This is why we construct the invariantization map only for this group parameter.

In the moving frame $\rho^{(1)}$, the component ε_5 follows from the normalization condition $u_x = 0$. This normalization condition is replaced by $u^{\rm d}_x = 0$ to guarantee the compatibility of the moving frame $\rho^\diamond$ with $\rho^{(1)}$ in the continuous limit. The problem when using the least square solution (8.2) to obtain $u^{\rm d}_x$ is that it will be very hard (or even impossible) to find the moving frame component ε_5 as the normalization procedure boils down to solving a high order polynomial equation for ε_5. This is why we use a less accurate approximation of $u^{\rm d}_x$ to compute ε_5. In particular, using the Taylor series expansions $u^{\rm r} = u + u^{\rm d}_x(x^{\rm r} - x)$ and $u^{\rm l} = u - u^{\rm d}_x(x - x^{\rm l})$ is sufficient to determine $u^{\rm d}_x$ at x, where $(x^{\rm r}, u^{\rm r})$ and $(x^{\rm l}, u^{\rm l})$ are the nodes lying immediately to the right and to the left of x. This leads to the usual centered difference approximation

$$u^{\rm d}_x = \frac{u^{\rm r} - u^{\rm l}}{x^{\rm r} - x^{\rm l}}. \tag{8.10}$$

Note that on a non-uniform grid it is not guaranteed that this approximation is second order accurate. However, it is numerically verified in the following section that the invariant meshless scheme which follows from the moving frame that employs the approximation (8.10) for $u^{\rm d}_x$ is more accurate than the associated non-invariant meshless scheme (8.5).

Computing the $\rho^\diamond$-component ε_5 from the normalization $u^{\rm d}_x = 0$ using the approximation (8.10) leads to the equation

$$(1 - \varepsilon_5 x^{\rm r})^3 u^{\rm r} - (1 - \varepsilon_5 x^{\rm l})^3 u^{\rm l} = 0,$$

which has the solution

$$\varepsilon_5 = \frac{\sqrt[3]{u^{\rm r}} - \sqrt[3]{u^{\rm l}}}{x^{\rm r}\sqrt[3]{u^{\rm r}} - x^{\rm l}\sqrt[3]{u^{\rm l}}}. \tag{8.11}$$

It is readily verified that in the continuous limit $x^{\rm r} \to x$ and $x^{\rm l} \to x$, the above solution (8.11) converges to $\varepsilon_5 = u_x/(xu_x + 3u)$. This shows that the moving frames $\rho^\diamond$ and $\rho^{(1)}$ are indeed compatible.

The invariant meshless counterpart to the scheme (8.5) is given by

$$\frac{\iota(\hat{u}) - \iota(u)}{\Delta t} = -\frac{4}{3}\iota(u)^{-7/3}\iota(u^{\rm d}_x)^2 + \iota(u)^{-4/3}\iota(u^{\rm d}_{xx}), \tag{8.12a}$$

and

$$\frac{\iota(\hat{u}) - \iota(\check{u})}{2\Delta t} = -\frac{4}{3}\iota(u)^{-7/3}\iota(u^{\rm d}_x)^2 + \iota(u)^{-4/3}\iota(u^{\rm d}_{xx}). \tag{8.12b}$$

The invariantized discrete derivatives follow from solving Eq. (8.2) using the invariantized expressions for $(u^{(4)})^{\mathrm{d}}|_x$, S and b, which are, respectively,

$$\iota((u^{(4)})^{\mathrm{d}}|_x) = \begin{pmatrix} \iota(u) \\ \iota(u_x^{\mathrm{d}}) \\ \iota(u_{xx}^{\mathrm{d}}) \\ \iota(u_{xxx}^{\mathrm{d}}) \end{pmatrix}, \quad \iota(S) = \begin{pmatrix} 1 & \iota(\Delta x^1) & \cdots & \frac{1}{24}\iota(\Delta x^1)^4 \\ 1 & \iota(\Delta x^2) & \cdots & \frac{1}{24}\iota(\Delta x^2)^4 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \iota(\Delta x^k) & \cdots & \frac{1}{24}\iota(\Delta x^k)^4 \end{pmatrix}, \quad \iota(b) = \begin{pmatrix} \iota(u^1) \\ \iota(u^2) \\ \vdots \\ \iota(u^k) \end{pmatrix}.$$

Likewise, the weight matrix W is invariantized to give $\iota(W) = \mathrm{diag}(\exp(-\mu\,\iota(\Delta x^j)^2/r^2))$. In all these formulas, ε_5 is the moving frame parameter (8.11) and we have

$$\iota(x^j) = \frac{x^j}{1-\varepsilon_5 x^j}, \quad \iota(u^j) = (1-\varepsilon_5 x^j)^3 u^j, \quad \iota(\hat{u}) = (1-\varepsilon_5 x)^3 \hat{u}, \quad \iota(\check{u}) = (1-\varepsilon_5 x)^3 \check{u}.$$

8.4 Numerical tests

We compare the invariant scheme (8.12) against the non-invariant meshless scheme (8.5) by carrying out numerical tests with the following three exact solutions of Eq. (8.4),

$$\begin{aligned} u_1 &= (2c_1 x - 3c_1^2 t + c_2)^{-3/4}, \\ u_2 &= \left(\frac{(x+c_1)^2}{t+c_2} + c_3(t+c_2)^2\right)^{-3/4}, \\ u_3 &= (c_1 x + c_2)^{-3}, \end{aligned} \tag{8.13}$$

where c_1, c_2, c_3 are arbitrary constants. The third solution is a stationary solution. For these and further solutions of Eq. (8.4), see [1, 20].

In all the numerical experiments reported we solve Eq. (8.4) with the invariant meshless scheme (8.12) and the non-invariant meshless scheme (8.5). We carry out the numerical integration on the interval $L = [1, 2]$ and choose the constants c_1, c_2 and c_3 so that the respective exact solution u_{e} is not singular within L. On this interval, we first create an equally-spaced grid with $N = 40$ grid points. Each of the grid point is then perturbed by adding a Gaussian distributed random number with zero mean and standard deviation $0.1 \cdot \Delta x$, where Δx is the spacing of the initial uniform grid. Dirichlet boundary conditions are used with the values of u^j at the boundaries given by the corresponding values of the exact solution u_1, u_2 or u_3.

Ten independent integrations using ten different realizations of the above described grid generation procedure are carried out for 1000 time steps of the size $\Delta t = 0.001$. The time step is rather small so as to avoid numerical instability in the course of the integration. Larger time steps could be used if implicit schemes would be employed instead of the explicit schemes in (8.12) and (8.5). The resulting root mean square errors

$$\mathrm{rmse} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(u_{\mathrm{n}}^j - u_{\mathrm{e}}^j)^2},$$

are computed after each integration, where u_{n}^j and u_{e}^j are the numerical and exact solutions of Eq. (8.4) at $t = 1$ at the nodal point x^j, respectively. The averaged root mean square errors for each of the three solutions (8.13) for the non-invariant and the invariant scheme are reported in Table 8.1 and denoted by $\mathrm{rmse}_{\mathrm{nis}}$ and $\mathrm{rmse}_{\mathrm{is}}$, respectively.

Table 8.1: Root mean square errors for the exact solutions (8.13) of the nonlinear diffusion equation (8.4).

	u_1	u_2	u_3
rmse_{nis}	$1.43 \cdot 10^{-5}$	$3.48 \cdot 10^{-5}$	$5.20 \cdot 10^{-6}$
rmse_{is}	$3.31 \cdot 10^{-6}$	$7.86 \cdot 10^{-7}$	0
$\text{rmse}_{\text{is}}/\text{rmse}_{\text{nis}} \cdot 100$	23.2%	2.3%	0%

In the first run (solution u_1), $c_1 = c_2 = 0.1$, in the second run (solution u_2), $c_1 = c_3 = 0$ and $c_2 = 10$ and $c_1 = c_2 = 0.1$ in the third run (solution u_3). It is worthwhile pointing out that the weight matrices W were chosen differently for the invariant and the non-invariant scheme. The reason for this is that the invariantization of Δx^j enters the weight matrix of the invariant scheme. To facilitate the comparison of the results, we have specified the parameter μ in each of the runs of the invariant scheme so that the entries in the weight matrices of both the invariant and the non-invariant scheme are of the same order of magnitude.

Table 8.1 shows that the invariant scheme is able to better approximate the exact solution at $t = 1$ in all three test cases. On top of that, for the case of the stationary solution u_3 we found that the invariant scheme approximates the exact solution up to machine precision.

As a further sensitivity test we run several integrations of the invariant scheme (8.12) and the non-invariant scheme (8.5), respectively, and vary the parameter μ in the weight matrices W. This parameter controls the influence of distant grid points in the meshless approximation of the discrete derivatives of u at the center node. We use u_1 as the exact solution in these runs. In Fig. 8.1 we depict the result of this sensitivity study (using $r = 0.2$). As was discussed before, it is necessary to choose μ differently in the invariant and non-invariant integrations due to the different magnitudes of Δx^j and $\iota(\Delta x^j)$. It can be seen from Fig. 8.1 that the invariant numerical scheme (8.12) gives (substantially) better integration results over virtually all values of the parameter μ.

The final sensitivity test we carry out is with respect to the parameter r, i.e. the radius within which grid points are used to compute the meshless approximations of the partial derivatives of u at the node x. The result of this study using u_1 as the exact solution is depicted in Fig. 8.2 (using the values for μ_{nis} and μ_{is} that were found to be optimal in the previous test for $r = 0.2$). Again, varying r the invariant scheme is (substantially) better than the non-invariant scheme. Moreover, for the lowest value of r chosen the non-invariant scheme (8.5) did not converge, whereas the invariant scheme (8.12) produced one of the best root mean square errors of all the integrations. This means that fewer nodal points are needed to compute a stable approximation of the derivatives for the invariant scheme in the present example.

Remark 8.2. We have also carried out numerical simulations with the schemes (8.12) and (8.5) computing $\iota((u^{(3)})^{\mathrm{d}}|_x)$ rather than $\iota((u^{(4)})^{\mathrm{d}}|_x)$ to approximate the first and second derivatives of u with respect to x in a less accurate way. Also in these simulations we found the invariant meshless scheme to be consistently better than the non-invariant discretization in approximating the above exact solutions.

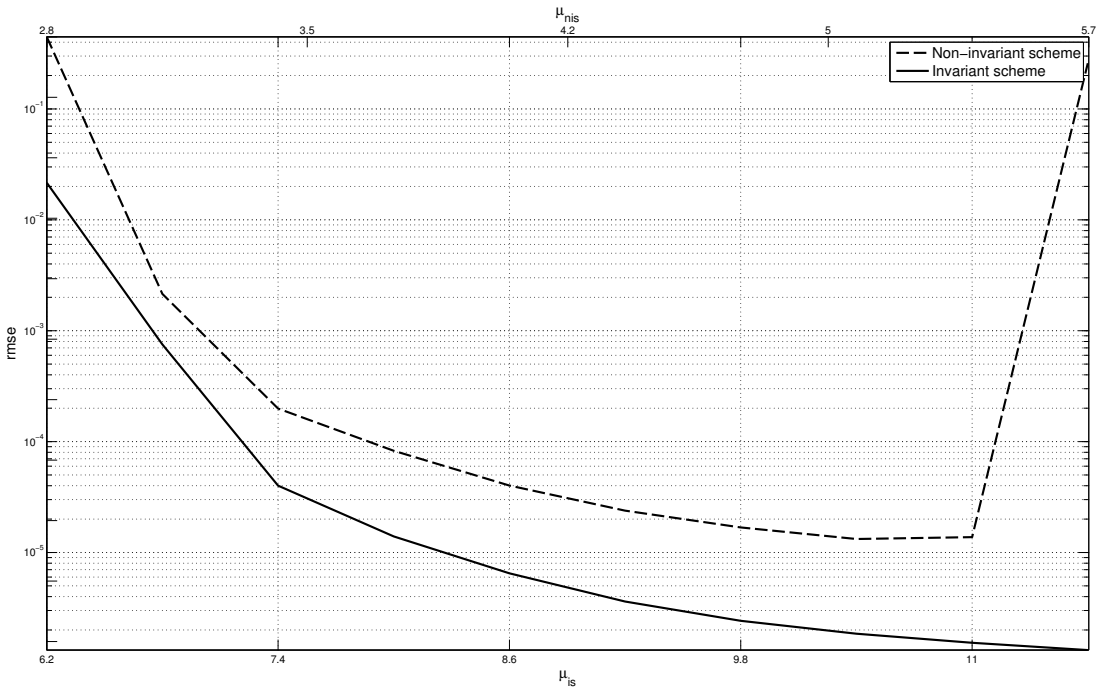

Figure 8.1: Sensitivity study of the invariant scheme (8.12) and the non-invariant scheme (8.5) with respect to the parameter μ in the weight matrix W. Upper x-axis: μ used in the invariant scheme. Lower x-axis: μ used in the non-invariant scheme.

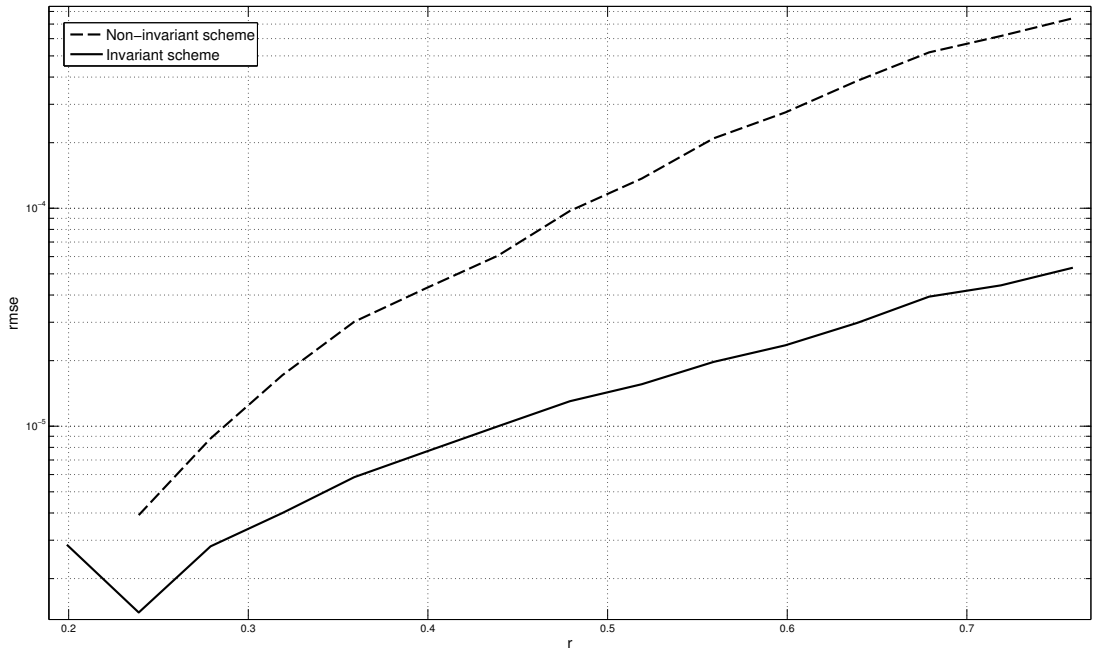

Figure 8.2: Sensitivity study of the invariant scheme (8.12) and the non-invariant scheme (8.5) with respect to the parameter r.

8.5 Conclusion

In this paper we developed a technique for the construction of invariant meshless discretization schemes. The key ingredient of this method is the application of the moving frame invariantization map to meshless discrete derivatives which ultimately boils down to the invariantization of a system of truncated Taylor series expansions.

Despite practically demonstrated solely for a one-dimensional evolution equation, the method of invariant meshless discretization as introduced in this paper is particularly suitable for multi-dimensional problems. Differential equations with more than one space dimensions posed a severe problem for the invariant scheme construction machinery available so far due to the necessity of using non-orthogonal, possibly moving discretization meshes. Consequently, most of what is known by today about invariant discretization schemes has been learned from the consideration of ordinary differential equations or single (1+1)-dimensional evolution equations [6, 8, 12, 13, 21]. It was only recently that methods for the construction of invariant numerical schemes for multi-dimensional systems of partial differential equations were developed, see e.g. [3, 4].

The construction of invariant meshless numerical integrators thus seems to be attractive for several reasons. From the point of view of invariant numerical schemes, it is beneficial to have one more method available that allows one to construct discretization schemes with symmetry properties for equations in any space dimension. In turn, from the side of meshless methods it is interesting to show that ideas from the field of geometric numerical integration can be successfully implemented into such methods. It was shown in this paper that the preservation of qualitative properties of a differential equation in a meshless approximation can substantially increase the quality of the scheme. The invariant discretization we constructed for a nonlinear diffusion equation is able to better reproduce several exact solutions of this equation in practically all the parameter ranges that can be tuned in the scheme. We thereby also demonstrated that preserving symmetries in a numerical integrator is not solely an academic problem.

It will be instructive to apply the proposed technique to multi-dimensional discretization problems and compare invariant meshless schemes against other types of invariant numerical schemes, both in terms of accuracy and computational cost. Also, certain symmetries (e.g. Galilean boosts) cannot be preserved on fixed discretization meshes. Moving grid points can lead to strongly distorted meshes and are thus likely to deteriorate the quality of the numerical solution or to slow down the convergence rate. This is what generally happens to Lagrangian integration schemes. As a matter of fact, most invariant numerical schemes preserving Galilean invariance are Lagrangian integrators [6, 8]. On the other hand, it is known that certain meshless methods are to some degree insensitive regarding the distribution of the nodes. It will therefore be informative to compare invariant meshless methods with discretizations that employ classical or invariant moving meshes such as those constructed in [4, 11].

Acknowledgements

This research was supported by the Austrian Science Fund (FWF), project J3182–N13. It is a pleasure to thank Professor Roman O. Popovych for helpful remarks on the manuscript. We gratefully acknowledge the useful remarks by the anonymous referees.

Bibliography

[1] Ames W.F., Anderson R.L., Dorodnitsyn V.A., Ferapontov E.V., Gazizov R.K., Ibragimov N.H. and Svirshchevskii S.R., *CRC handbook of Lie group analysis of differential equations. Vol. 1. Symmetries, exact solutions and conservation laws. Edited by N. H. Ibragimov*, CRC Press, Boca Raton, 1994.

[2] Belytschko T., Krongauz Y., Organ D., Fleming M. and Krysl P., Meshless methods: an overview and recent developments, *Comput. Methods Appl. Mech. Engrg.* **139** (1996), 3–47.

[3] Bihlo A. and Nave J.C., Invariant discretization schemes for the heat equation, 2012, arXiv:1209.5028, 21 pp.

[4] Bihlo A. and Popovych R.O., Invariant discretization schemes for the shallow-water equations, *SIAM J. Sci. Comput.* **34** (2012), B810–B839, arXiv:1201.0498.

[5] Ding H., Shu C., Yeo K. and Xu D., Development of least-square-based two-dimensional finite-difference schemes and their application to simulate natural convection in a cavity, *Comput. & Fluids* **33** (2004), 137–154.

[6] Dorodnitsyn V., *Applications of Lie Groups to Difference Equations*, vol. 8 of *Differential and integral equations and their applications*, Chapman & Hall/CRC, Boca Raton, FL, 2011.

[7] Dorodnitsyn V.A., Transformation groups in mesh spaces, *J. Sov. Math.* **55** (1991), 1490–1517.

[8] Dorodnitsyn V.A. and Kozlov R., A heat transfer with a source: the complete set of invariant difference schemes, *J. Nonlin. Math. Phys.* **10** (2003), 16–50.

[9] Fels M. and Olver P.J., Moving coframes: I. A practical algorithm, *Acta Appl. Math.* **51** (1998), 161–213.

[10] Fels M. and Olver P.J., Moving coframes. II. Regularization and theoretical foundations, *Acta Appl. Math.* **55** (1999), 127–208.

[11] Huang W. and Russell R.D., *Adaptive Moving Mesh Methods*, Springer, New York, 2010.

[12] Kim P., Invariantization of the Crank–Nicolson method for Burgers' equation, *Physica D* **237** (2008), 243–254.

[13] Levi D. and Winternitz P., Continuous symmetries of difference equations, *J. Phys. A* **39** (2006), R1–R63.

[14] Liszka T.J., Duarte C.A.M. and Tworzydlo W.W., Hp-meshless cloud method, *Comput. Methods Appl. Mech. Engrg.* **139** (1996), 263–288.

[15] Milewski S., Meshless finite difference method with higher order approximation—applications in mechanics, *Arch. Comput. Methods Eng.* **19** (2012), 1–49.

[16] Nguyen V.P., Rabczuk T., Bordas S. and Duflot M., Meshless methods: a review and computer implementation aspects, *Math. Comput. Simulation* **79** (2008), 763–813.

[17] Olver P.J., Geometric foundations of numerical algorithms and symmetry, *Appl. Algebra Engrg. Comm. Comput.* **11** (2001), 417–436.

[18] Olver P.J., Generating differential invariants, *J. Math. Anal. Appl.* **333** (2007), 450–471.

[19] Olver P.J. and Pohjanpelto J., Moving frames for Lie pseudo-groups, *Canadian J. Math.* **60** (2008), 1336–1386.

[20] Polyanin A.D. and Zaitsev V.F., *Handbook of Nonlinear Partial Differential Equations*, Chapman & Hall/CRC, New York, 2004.

[21] Rebelo R. and Valiquette F., Symmetry preserving numerical schemes for partial differential equations and their numerical tests, *J. Difference Equ. Appl.*, to appear, arXiv:1110.5921, 2012.

Acknowledgments

The research reported in this thesis was funded by the Austrian Academy of Sciences (through an APART Fellowship), the Austrian Science Fund (through a Schrödinger Fellowship), the Canada Research Chairs program (through a Tier II CRC in Numerical Analysis and Scientific Computing), the National Science and Engineering Research Council (through a Discovery Grant), and the Research and Development Cooperation of Newfoundland and Labrador.

I am deeply indebted to my co-authors, George Bluman, Xavier Coiteux-Roy, Elsa Maria Dos Santos Cardoso-Bihlo, Jean-Christophe Nave, Roman O. Popovych and Pavel Winternitz. It was (and is) a pleasure to work with them in this fascinating field of applied mathematics. In addition, I would like to thank my colleagues (and co-authors of papers not included in this monograph) and students (former and current), for providing me with the opportunity to work on a wide variety of problems.

Curriculum Vitae

Contact

Address	Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada A1C 5S7
E-mail	abihlo@mun.ca
Homepage	http://www.math.mun.ca/∼abihlo/

Professional experience

Nov 2015–now	*Tier II Canada Research Chair* in Numerical Analysis and Scientific Computing at the Department of Mathematics and Statistics, Memorial University of Newfoundland
Nov 2015–now	*Assistant professor* at the Department of Mathematics and Statistics, Memorial University of Newfoundland
Jan 2015–Oct 2015	*Visiting postdoctoral fellow* at the Department of Mathematics, University of British Columbia
Jan 2014–Oct 2015	*Postdoctoral fellow* at the Department of Mathematics and Statistics and the Department of Earth Sciences, Memorial University of Newfoundland
Aug 2013–Dec 2013	*Postdoctoral fellow* at the Department of Mathematics and Statistics, McGill University
June 2012–July 2013	*Visiting postdoctoral fellow* at the Department of Mathematics and Statistics, McGill University
Aug 2011–July 2013	*Postdoctoral fellow* at the Centre de recherches mathématiques, Université de Montréal
Jan 2011–July 2011	*Postdoctoral fellow* at the Faculty of Mathematics, University of Vienna
Jan 2009–Dec 2010	*Ph.D. fellowship* at the Faculty of Mathematics, University of Vienna
Oct 2008–Dec 2008	*Ph.D. position* at the Faculty of Mathematics, University of Vienna
Oct 2007–Sept 2008	*Research assistant* at the Department of Meteorology and Geophysics, University of Vienna

Academic education

2003–2004	Study of Meteorology and Philosophy at the University of Innsbruck
2004–2007	Study of Meteorology at the University of Vienna
11 Jun 2007	Degree *Magister der Naturwissenschaften* (M.Sc.) in Meteorology, with distinction
2008–2010	Ph.D. studies in Meteorology at the University of Vienna
7 Dec 2010	Defense of Ph.D. thesis at the University of Vienna, passed with distinction
2014–2018	Habilitation program of the Austrian Academy of Sciences in Applied Mathematics at the University of Vienna

Theses

- A. Bihlo, 2010. *Symmetry methods in the atmospheric sciences.* Ph.D. thesis, University of Vienna, 165 pp. (Supervisor: Prof. Michael Hantel)
- A. Bihlo, 2007. *Solving the vorticity equation with Lie groups.* M.Sc. thesis, University of Vienna, 78 pp. (Supervisor: Prof. Michael Hantel)

Grants and awards

- CAIMS–PIMS Early Career Award 2018
- NSERC Discovery Grant 2016–2021 (CAD 115,000)
- Research & Development Corporation of Newfoundland and Labrador (RDC) IgniteR&D Program 2016–2018 (CAD 98,000)
- Hibernia Geophysics Fund 2016–2018 (CAD 18,000)
- Canada Foundation for Innovation (CFI) John R. Evans Leaders Fund (JELF) Partnership program 2016 (CAD 50,000)
- RDC CFI LeverageR&D program 2016 (CAD 88,515)
- RDC CRC LeverageR&D program 2015–2020 (CAD 100,000)
- Tier II Canada Research Chair Program *Innovative methods for geometric numerical integration* 2015–2020 (CAD 500,000)
- PIMS Postdoctoral Fellowship 2014–2016 (CAD 80,000 reduced to CAD 10,000 to avoid double funding)
- APART Fellowship of the Austrian Academy of Sciences 2014–2018 (200,000 EUR)
- Würdigungspreis des Bundesmininsteriums für Wissenschaft und Forschung (Award of Excellence of the Austrian Ministry for Science and Research) in 2011 (2,500 EUR)
- Schrödinger Fellowship of the Austrian Science Fund (FWF) 2011–2014 (145,000 EUR)
- Awarded the doctor's degree *Sub Auspiciis Praesidentis* in March 2011 (highest distinction in Austria; awarded to 3 out of 642 Ph.D. graduates at the University of Vienna in 2011)
- DOC Fellowship of the Austrian Academy of Sciences 2009–2010 (60,000 EUR)
- Merit scholarships of the University of Vienna in 2005, 2006, 2007 (700 EUR each)

Research projects

April 2016–March 2021	NSERC Discovery Grant *Geometric foundation of invariant and conservative parameterization schemes*
Nov 2015–Oct 2020	Tier II Canada Research Chair in *Numerical Analysis and Scientific Computing*
May 2014–Dec 2018	Habilitation fellowship *Geometric methods of discretization and parameterization*
May 2016–April 2018	RDC IgniteR&D Program *Meshless numerical integrators for the improvement of geophysical electromagnetic exploration methods*
April 2013–March 2018	Co-author and member of FWF project *Extended group analysis of differential equations* (P.I. Prof. Dr. Roman O. Popovych)
Aug 2011–July 2014	Postdoctoral fellowship *Symmetries in atmospheric numerical models*
Feb 2010–Feb 2014	Austrian representative of COST project *Basic Concepts for Convection Parameterization in Weather Forecast and Climate Models* (P.I. Dr. Jun-Ichi Yano)
July 2008–June 2013	Co-author and member of FWF project *Classification problems of group analysis* (P.I. Prof. Dr. Roman O. Popovych)
April 2009–Oct 2012	Co-author and member of FWF project *Nambu calculus in dynamic meteorology* (P.I. Prof. Dr. Michael Hantel)
Jan 2009–Dec 2010	Ph.D. fellowship *Symmetry methods in dynamic meteorology*

Publications

Published papers

1. A. Bihlo and S. MacLachlan, 2018. Well-balanced mesh-based and meshless schemes for the shallow-water equations *BIT Numerical Mathematics* **58** (3), 579–598, arXiv:1702.07749.
2. R. Brecht, A. Bihlo, S. MacLachlan and J. Behrens, 2018. A well-balanced meshless tsunami propagation and inundation model, *Advances in Water Resources* **115**, 273–285, arXiv:1705.09831.
3. A. Wan, A. Bihlo and J.-C. Nave, 2017. Conservative methods for dynamical systems, *SIAM Journal on Numerical Analysis* **55** (5), 2255–2285, arXiv:1612.02417.
4. S. Opanasenko, A. Bihlo and R.O. Popovych, 2017. Group analysis of general Burgers–Korteweg–de Vries equations. *Journal of Mathematical Physics* **58**, 081511 (37 pages), arXiv:1703.06932.
5. A. Bihlo, C.G. Farquharson, R.D. Haynes, J.C. Loredo-Osti, 2017. Stochastic domain decomposition for the solution of the two-dimensional magnetotelluric problem. *Computational Geosciences* **21** (1), 117–129, arXiv:1603.09311.
6. A. Bihlo and R.O. Popovych, 2017. Group classification of linear evolution equations. *Journal of Mathematical Analysis and Applications* **448** (2), 982–1005, arXiv:1605.09251.

7. A. Wan, A. Bihlo and J.-C. Nave, 2016. The multiplier method to construct conservative finite difference schemes for ordinary and partial differential equations. *SIAM Journal on Numerical Analysis* **54** (1), 86–119, arXiv:1411.7720.

8. A. Bihlo and R.D. Haynes, 2016. A stochastic domain decomposition method for time dependent mesh generation. In *Springer Lecture Notes in Computational Sciences and Engineering*, 107–115, arXiv:1402.0266.

9. A. Bihlo, R.D. Haynes and E.J. Walsh, 2015. Stochastic domain decomposition for time dependent adaptive mesh generation. *Journal of Mathematical Study*, **48** (2), 106–124, arXiv:1504.00084.

10. A. Bihlo, E.M. Dos Santos Cardoso-Bihlo and R.O. Popovych, 2015. Algebraic method for finding equivalence groups. *Journal of Physics: Conference Series* **621**, 012001 (17 pp), arXiv:1503.06487.

11. A. Bihlo, X. Coiteux-Roy and P. Winternitz, 2015. Invariant discretization schemes for the Korteweg–de Vries equation. *Journal of Physics A: Mathematical and Theoretical* **48** (5), 055201 (25 pages), arXiv:1409.4340.

12. A. Bihlo and R.D. Haynes, 2014. Parallel stochastic methods for PDE based grid generation. *Computers and Mathematics with Applications* **68** (8), 804–820, arXiv:1310.3435.

13. A. Bihlo and J.-C. Nave, 2014. Convecting reference frames and invariant numerical models. *Journal of Computational Physics* **271**, 656–663 arXiv:1301.5955.

14. A. Bihlo, E.M. Dos Santos Cardoso-Bihlo and R.O. Popovych, 2014. Invariant parameterization and turbulence modeling on the beta-plane. *Physica D* **269**, 48–62, arXiv:1112.1917.

15. S. Szatmari and A. Bihlo, 2014. Symmetry analysis of a system of modified shallow-water equations. *Communications in Nonlinear Science and Numerical Simulation* **19** (3), 530-537, arXiv:1212.5823.

16. A. Bihlo and J.-C. Nave, 2013. Invariant discretization schemes using evolution–projection techniques. *SIGMA* **9**, 052 (23 pages), arXiv:1209.5028.

17. A. Bihlo and G. Bluman, 2013. Conservative parameterization schemes. *Journal of Mathematical Physics* **54** (8), 083101 (24 pages), arXiv:1209.4279.

18. A. Bihlo, 2013. Invariant meshless discretization schemes. *Journal of Physics A: Mathematical and Theoretical* **46** (6), 062001 (12 pp), arXiv:1210.2762.

19. A. Bihlo and R.O. Popovych, 2012. Invariant discretization schemes for the shallow-water equations. *SIAM Journal on Scientific Computing* **34** (6), B810–B839, arXiv:1201.0498.

20. A. Bihlo, E.M. Dos Santos Cardoso-Bihlo and R.O. Popovych, 2012. Complete group classification of a class of nonlinear wave equations. *Journal of Mathematical Physics* **53**, 123515, 32 pp, arXiv:1106.4801.

21. R.O. Popovych and A. Bihlo, 2012. Symmetry preserving parameterization schemes. *Journal of Mathematical Physics* **53** (7), 073102, 36 pp, arXiv:1010.3010.

22. A. Bihlo and R.O. Popovych, 2012. Lie reduction and exact solutions of the vorticity equation on the rotating sphere. *Physics Letters A* **376** (14), 1179–1184, arXiv:1112.3019.

23. A. Bihlo and R.O. Popovych, 2011. Lie symmetry analysis and exact solutions of the quasi-geostrophic two-layer problem. *Journal of Mathematical Physics* **52** (3), 033103, 24 pp, arXiv:1010.1542.

24. E.M. Dos Santos Cardoso–Bihlo, A. Bihlo and R.O. Popovych, 2011. Enhanced preliminary group classification of a class of generalized diffusion equations. *Communications in Nonlinear Science and Numerical Simulation* **16** (9), 3622-3638, arXiv:1012.0297.

25. A. Bihlo and J. Staufer, 2011. Minimal atmospheric finite-mode models preserving symmetry and generalized Hamiltonian structures, *Physica D* **240** (7), 599–606, arXiv:0909.1957.

26. A. Bihlo and R.O. Popovych, 2009. Lie symmetries and exact solutions of the barotropic vorticity equation. *Journal of Mathematical Physics* **50** (12), 123102, 12 pp, arXiv:0902.4099.

27. A. Bihlo and R.O. Popovych, 2009. Symmetry justification of Lorenz' maximum simplification. *Nonlinear Dynamics*, **61** (1), 101–107, arXiv:0805.4061.

28. A. Bihlo and R.O. Popovych, 2009. Symmetry analysis of barotropic potential vorticity equation. *Communications in Theoretical Physics*, **52** (4), 697–700, arXiv:0811.3008.

29. A. Bihlo, 2008. Rayleigh-Bénard Convection as a Nambu–metriplectic problem. *Journal of Physics A: Mathematical and Theoretical*, **41** (29), 292001, 6 pp, arXiv:0803.4458.

Peer-reviewed conference proceedings

1. A. Bihlo and R.O. Popovych, 2011. Point symmetry group of the barotropic vorticity equation, *Proceedings of the 5th Workshop "Group Analysis of Differential Equations & Integrable Systems"* (June 6–10, 2010, Protaras, Cyprus), 15–27, arXiv:1009.1523.

2. A. Bihlo, 2009. Symmetries in atmospheric sciences, *Proceedings of the 4th Workshop "Group Analysis of Differential Equations & Integrable Systems"* (October 26–30, 2008, Protaras, Cyprus), 6–12, arXiv:0902.4112.

Peer-reviewed book chapters

1. A. Bihlo and F. Valiquette, 2017. Symmetry-preserving numerical schemes, in *Symmetries and Integrability of Difference Equations*, 261–324, Springer, arXiv:1608.02557.

2. A. Bihlo, E.M. Dos Santos Cardoso-Bihlo and R.O. Popovych, 2015. Invariant and conservative parameterization schemes, in volume 2 of *Parameterization of Atmospheric Convection*, (R.S. Plant and J.I. Yano, Eds.), World Scientific, Imperial College Press, London.

Submitted papers

1. R. Brecht, W. Bauer, A. Bihlo, F. Gay-Balmaz, S. MacLachlan, 2018. Variational integrators for the rotating shallow-water equations on the sphere, arXiv:1808.10507.

2. A. Bihlo and F. Valiquette, 2018. Symmetry-preserving finite element schemes: An introductory investigation, arXiv:1803.10058.

3. F. Donzelli, A. Bihlo, M. Kischinhevsky, C.G. Farquharson, 2017. Massively parallel stochastic solution of the geophysical gravity problem, arXiv:1709.07469.

4. S. Opanasenko, A. Bihlo, R.O. Popovych and A. Sergyeyev, 2017. Extended symmetry analysis of isothermal no-slip drift flux model, arXiv:1705.09277.

5. R.O. Popovych and A. Bihlo, 2017. Inverse problem on conservation laws, arXiv:1705.03547.

Talks and presentations

Invited talks

- Oct 2018: "Invariant parameterization in geophysical fluid dynamics", talk at the School of Mathematics and Statistics, University of St Andrews
- June 2018: "Geometry-preserving numerical modeling in geophysical fluid dynamics", plenary talk at CAIMS 2018, Toronto
- June 2018: "A meshless tsunami propagation and inundation model", talk in session *Computational Mathematics and Applications* at CAIMS 2018, Toronto
- May 2017: "A well-balanced meshless tsunami propagation model", NCAR/IMAGe Theme of the Year 2017 – Workshop on Multiscale Geoscience Numerics, Boulder, CO *(cancelled due to illness)*
- June 2016: "Symmetry-preserving discretization schemes", talk at the ASIDE conference, Montreal
- May 2016: "Conservative discretization schemes", talk at the Institute of Meteorology and Geophysics, University of Vienna
- May 2015: "Novel methods for geometric numerical integration", talk at the Department of Mathematics and Statistics, University of Saskatchewan
- Oct 2014: "Geometry-preserving numerical modeling", talk at the Department of Mathematics and Statistics, Memorial University of Newfoundland
- Feb 2013: "Conservative parameterization schemes", talk at the Institute of Meteorology and Geophysics, University of Vienna
- Feb 2013: "Invariant discretization schemes", talk at the Institute of Meteorology and Geophysics, University of Vienna
- Feb 2013: "Geometry preserving modeling in geophysical fluid dynamics", talk at the Department of Mathematics and Statistics, University of Reading
- Dec 2012: "Invariant subgrid-scale closure schemes for turbulence modeling", talk in 2012 CMS Winter Meeting session *Applied mathematics*, Montréal
- Dec 2012: "Invariant discretization schemes", talk in 2012 CMS Winter Meeting session *Symmetries of differential and difference equations*, Montréal
- June 2012: "Invariant parameterization schemes and turbulence modeling", talk at the Department of Mathematics, University of British Columbia
- June 2012: "Invariant turbulence modeling", talk at the Center for Turbulence Research, Stanford University
- May 2011: "A tutorial on Hamiltonian mechanics", talk at the COST working group meeting at the Institute of Meteorology, University of Munich
- Sep 2010: "A method for constructing exact solutions of partial differential equations", talk at the Faculty of Sciences, University of Lisbon
- Jan 2009: "Symmetry methods in dynamic meteorology", talk at the Department of Meteorology, University of Frankfurt
- July 2008: "Symmetrien und einige Anwendungen in der dynamischen Meteorologie", talk at the Department of Meteorology, University of Bonn

Selected other presentations

- April 2018: "Exactly conservative discretization schemes for dynamical systems", talk at the Department of Mathematics and Statistics, Memorial University of Newfoundland.
- March 2018: "Invariant discretization beyond finite differences", talk at the Department of Mathematics and Statistics, Memorial University of Newfoundland.
- Jan 2018: "A well-balanced meshless tsunami propagation and inundation model", talk at the Department of Mathematics, University of British Columbia
- Dec 2016: "Well-balanced mimetic mesh-based and meshless schemes for the shallow-water equations with bottom topography", talk at the 2016 CMS Winter Meeting, Niagara Falls
- Mar 2015: "Recent advancements in geometric numerical integration", talk at the Department of Mathematics, University of British Columbia
- Mar 2015: "Stochastic domain decomposition for parallel grid generation", talk at the Institute of Applied Mathematics, University of British Columbia
- Feb 2015: "Invariant and conservative numerical schemes: Theory and applications", talk at the Department of Mathematics, University of British Columbia
- Sep 2014: "Domain decomposition and flux solution of the 2d MT problem", talk at the Department of Earth Sciences, Memorial University of Newfoundland
- June 2014: "Invariant discretization schemes", talk at the XXIInd International Conference on Integrable Systems and Quantum symmetries, Prague, Czech Republic
- June 2014: "Grid generation using stochastic domain decomposition", talk at the Institute of Meteorology and Geophysics, University of Vienna
- Nov 2013: "Grid generation using probabilistic and deterministic methods", talk at the Department of Mathematics and Statistics, McGill University
- June 2013: "Invariant discretization schemes", talk in the "1st Canadian Symposium in Numerical Analysis and Scientific Computing", CAIMS 2013, Quebec City, QC, Canada
- Dec 2012: "Complete group classification of a class of nonlinear wave equations", poster presentation at the 2012 CMS Winter Meeting, Montréal
- Nov 2012: "Invariant discretization of partial differential equations", talk at the Department of Mathematics and Statistics, McGill University
- May 2012: "Invariant parameterization schemes", poster presentation at the conference *Symmetries of Differential Equations: Frames, Invariants and Applications*, University of Minnesota
- May 2012: "Invariant turbulence modeling", poster presentation at the conference *Symmetries of Differential Equations: Frames, Invariants and Applications*, University of Minnesota
- Sept 2011: "Invariant parameterization schemes", poster presentation at the workshop *Balance, Boundaries and Mixing in the Climate Problem*, Université de Montréal
- Sept 2011: "Invariant discretization schemes for the shallow-water equations", talk at the Centre de recherches mathématiques, Université de Montréal

- Dec 2010: "Symmetry methods in the atmospheric sciences", defense talk at the Faculty of Geosciences, Geography and Astronomy, University of Vienna
- June 2010: "Symmetry preserving parameterization schemes", talk at the 5^{th} International Workshop *Group Analysis of Differential Equations and Integrable systems*, Protaras, Cyprus
- May 2010: "Symmetry preserving discretization schemes", EGU poster session AS1.5 *Recent developments in geophysical fluid dynamics*, Vienna
- Nov 2009: "Der Nambu Kalkül in der dynamischen Meteorologie", poster session at the 3^{rd} *Austrian Meteorology Day*, Graz, Austria
- April 2009: "Symmetry methods in dynamic meteorology", talk in EGU session AS1.13 *Recent developments in geophysical fluid dynamics*, Vienna
- July 2008: "Symmetry methods in dynamic meteorology", talk in session *Symbolic Symmetry Analysis and Its Applications*, ACA 2008 conference, Hagenberg, Austria
- April 2008: "Application of finite and infinite dimensional Nambu mechanics in dynamic meteorology", EGU poster session AS1.11 *Recent developments in geophysical fluid dynamics*, Vienna
- Dec 2007: "Nambu Mechanik. Vom Kreisel zum Chaos", talk at the Department of Meteorology, Freie Universität Berlin
- Dec 2006: "Can Lie groups improve weather forecasts?", talk at the Faculty of Mathematics, University of Vienna

Conference co-organization

9–13 Sept 2019	*Variational Discretization for Geophysical Fluid Dynamics*, Fields Institute, Toronto, ON, Canada
7–10 Dec 2018	*Advances and Applications in Geometric and Structure-Preserving Discretizations*, 2018 CMS Winter Meeting, Vancouver, BC, Canada
23–27 July 2018	*DD25 Poster Session*, 25th International Domain Decomposition Conference, St. John's, NL, Canada
21–18 Feb 2018	*Variational Integration on the Sphere*, Workshop, Reykjavik, Iceland
17–21 July 2017	*Third Canadian Symposium in Numerical Analysis and Scientific Computing*, Annual Meeting of the Canadian Applied and Industrial Mathematics Society, Halifax, NS, Canada
11–16 June 2017	*Connections in Geometric Numerical Integration and Structure-Preserving Discretization*, Banff International Research Station, Banff, AB, Canada
2–5 Dec 2016	*Recent Advances in Structure-Preserving Discretizations*, 2016 CMS Winter Meeting, Niagara Falls, ON, Canada

Conference attendances and short-term scientific visits

4–7 June 2018	*CAIMS 2018*, Annual Meeting of the Canadian Applied and Industrial Mathematics Society, Toronto, ON, Canada
22–29 Jan 2018	Research stay at the Department of Mathematics at the University of British Columbia, Vancouver, BC, Canada
15–19 Aug 2017	Research stay at the Department of Mathematics and Statistics, McGill University, Montreal, QC, Canada
5–8 July 2017	Research stay at Imperial College London, London, UK
16 April–10 May 2017	Research stay at the Faculty of Mathematics, University of Vienna, Vienna, Austria
16–26 Aug 2016	Research stay at the Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada
3–9 July 2016	*SIDE 12 – Symmetries and Integrability of Difference Equations*, Sainte-Adele, QC, Canada
27 June–1 July 2016	*ASIDE – Abecedarian of SIDE*, Montreal, QC, Canada
10 May–4 June 2016	Research stay at the Institute of Meteorology and Geophysics, University of Vienna, Vienna, Austria
4–8 May 2015	Research stay at the Department of Mathematics and Statistics at the University of Saskatchewan, Saskatoon, SK, Canada
17–22 Aug 2014	*Recent Developments in Adaptive Methods for PDEs*, Collaborative Workshop and Short Course, Memorial University of Newfoundland, St. John's, NL, Canada
19–26 July 2014	Research stay at the Centre de recherches mathématiques at the Université de Montréal, Montreal, QC, Canada
26–27 June 2014	*XXIInd International Conference on Integrable Systems and Quantum symmetries* (ISQS-22), Prague, Czech Republic
3–31 Aug 2013	Research stay at the Department of Mathematics at the University of British Columbia, Vancouver, BC, Canada
16–20 June 2013	*CAIMS 2013*, Annual Meeting of the Canadian Applied and Industrial Mathematics Society, Quebec City, QC, Canada
24 May–14 June 2013	Research stay at the Department of Mathematics and Statistics at Memorial University of Newfoundland, St. John's, NL, Canada
12 Feb–14 Feb 2013	Invited lecture series *Nambu calculus and Lie groups in theoretical meteorology*, University of Vienna, Austria
7 Dec–10 Dec 2012	2012 CMS Winter Meeting, Montréal, QC, Canada
21 May–14 June 2012	Research stay at the Department of Mathematics at the University of British Columbia, Vancouver, BC, Canada
17–20 May 2012	Conference *Symmetries of Differential Equations: Frames, Invariants and Applications*, University of Minnesota, MN, USA

28–30 Sept 2011	Workshop *Balance, Boundaries and Mixing in the Climate Problem*, Université de Montréal, Montréal, QC, Canada
3 May 2011	COST working group meeting, Munich, Germany
22–25 March 2011	COST workshop, Cambridge, UK
Sept 2010	Research stay at the Centro de Matemática e Aplicações at the Instituto Superior Técnico, Lisbon, Portugal
6–10 June 2010	Fifth International Workshop *Group Analysis of Differential Equations and Integrable Systems*, Protaras, Cyprus
5 May 2010	European Geosciences Union – General Assembly, Vienna, Austria
6 Nov 2009	Third Austrian Meteorology Day, Graz, Austria
15–19 June 2009	Kickoff-Workshop *Nambu Calculus*, Vienna, Austria
23 April 2009	European Geosciences Union – General Assembly, Vienna, Austria
27–28 July 2008	Applications of Computer Algebra (ACA) – RISC Summer 2008, Hagenberg, Austria
17 April 2008	European Geosciences Union – General Assembly, Vienna, Austria
10–14 Dec 2007	Research stay at the Department of Meteorology at the Freie Universität Berlin, Germany

Students

Postdoctoral fellows

- Dr. Hormoz Jahandari, *Mimetic discretization for Maxwell's equations*, Department of Mathematics and Statistics, Memorial University of Newfoundland, November 2018–now
- Dr. Fabrizio Donzelli, *Deterministic and stochastic domain decomposition for Maxwell's equations*, Department of Mathematics and Statistics, Memorial University of Newfoundland, May 2016–now

Ph.D. students

- Stanislav Opanasenko, *Modern developments in group classification of differential equations*, Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2017–now
- Rüdiger Brecht, *Meshless methods for tsunami modeling*, joint with Professor Scott MacLachlan, Department of Mathematics and Statistics, Memorial University of Newfoundland, May 2017–now

MSc students

- Lada Atamanchuk-Anhel, *Invariant and conservative parameterization schemes*, Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2017–now

- Nataliia Poltavets, *The algebraic method for finding discrete symmetries*, Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2017–now
- Evan Kielley, *Development of an iceberg ensemble prediction model*, joint with Professor Scott MacLachlan and Professor James Munroe, Department of Physics and Physical Oceanography, Memorial University of Newfoundland, May 2017–now
- Oleksandr Abramov, *Stochastic domain decomposition for parallel grid generation*, joint with Professor Ronald D. Haynes, Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2016–December 2018
- Stanislav Opanasenko, *Extended symmetry analysis of isothermal no-slip drift flux model*, Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2016–August 2017
- Rüdiger Brecht, *Solution of the shallow-water equations with wetting and drying boundary conditions through RBF discretisation*, joint with Professor Jörn Behrens (University of Hamburg), Department of Mathematics and Statistics, Memorial University of Newfoundland, September 2016–April 2017
- Nelson Feyeux, *Invariant numerical schemes for partial differential equations*, joint with Professor Jean-Christophe Nave, Department of Mathematics and Statistics, McGill University, April 2013–September 2013

Undergraduate students

- Ben Morrison, *Parallel implementation of a meshless tsunami model* (NSERC Undergraduate Summer Research Award), Department of Mathematics and Statistics, Memorial University of Newfoundland, April–August 2017.
- Leah Genge, *Stochastic domain decomposition for parallel grid generation* (honors thesis), joint with Professor Ronald D. Haynes, Department of Mathematics and Statistics, Memorial University of Newfoundland, May 2016–April 2017
- Xavier Coiteux–Roy, *Invariant discretization schemes for the Korteweg–de Vries equation* (undergraduate summer research project), joint with Professor Pavel Winternitz, Centre de recherches mathématiques, Université de Montréal, May–August 2013
- Simon Szatmari, *Invariant discretization of the Monge–Ampère equation* (honors thesis), joint with Professor Jean-Christophe Nave, Department of Mathematics and Statistics, McGill University, September 2012–April 2013

Teaching experience

Memorial University

- CMSC 6920 – Scientific programming (2017, 2018)
- MATH 3132 – Numerical analysis I (2016, 2018)
- MATH 3202 – Vector calculus (2017)
- MATH 6201 – Numerical methods for time-dependent differential equations (2016, 2017)
- MATH 6210 – Numerical methods for differential equations (2018)
- MATH 6261 – Geometric numerical integration (2017)

McGill University

- MATH 262 – Intermediate calculus (2013)

University of Vienna

- Fluid mechanics (2009)
- Geophysical fluid dynamics (2008, 2016)
- Numerical and mathematical methods in meteorology (2007, 2009)
- Programming in Matlab (2008)
- Selected topics of dynamical meteorology (2011)

Skills

Language skills	German (native), English (fluent)
Programming skills	Matlab, Python, C/C++, Maple
Software skills	Illustrator, Photoshop, LaTeX, Office, Linux, Windows

Referee and reviewer

Acta Applicandae Mathematicae, *Applied Mathematics and Computation*, *Atmósfera*, *Communications in Nonlinear Science and Numerical Simulation*, *Entropy*, *European Journal of Applied Mathematics*, *Europhysics Letters*, *Foundations of Computational Mathematics*, *Journal of Computational Physics*, *Journal of Geometry and Physics*, *Journal of Mathematical Analysis and Applications*, *Journal of Mathematical Physics*, *Journal of Mathematical Study*, *Journal of Physics A*, *Mathematical Review*, *Meccanica*, *Nonlinear Dynamics*, *Numerical Mathematics: Theory, Methods and Applications*, *Physica A*, *Physica D*, *Physica Scripta*, *SIAM Journal on Scientific Computing*, *SIGMA "Symmetry, Integrability and Geometry: Methods and Applications"*