

**Numerical Methods for Fluid Flow:  
High Order SBP Schemes, IMEX Advection-Diffusion Splitting and  
Positivity Preservation for Production-Destruction-PDEs**

Habilitationsschrift

eingereicht im Fachbereich Mathematik und Naturwissenschaften  
der Universität Kassel

vorgelegt von  
**Sigrun Ortleb**  
aus Görlitz

Kassel, im Januar 2021



# Contents

## Introduction

v

<b>1 High Order Schemes for Conservation Laws in Fluid Dynamics</b>	
<i>Benefit of SBP Properties in Modern Space Discretization Schemes</i>	<b>1</b>
1.1 The SBP property: Historical background and generalizations . . . . .	5
1.2 The DG scheme in SBP framework . . . . .	18
1.2.1 The one-dimensional case . . . . .	18
1.2.2 The subcell finite volume property . . . . .	24
1.2.3 The DG scheme with numerical fluxes in upwind SBP framework . . . .	28
1.2.4 Extension to 2D cartesian grids via tensor-product SBP operators . .	33
1.2.5 Extension to DG schemes on triangular grids . . . . .	36
1.3 Energy stability of flux reconstruction schemes . . . . .	42
1.4 Kinetic energy preserving DG schemes for the Euler- and Navier-Stokes equations	48
1.4.1 Kinetic energy preservation in one space dimension . . . . .	50
1.4.2 Numerical experiments: accuracy and evolution of kinetic energy . .	61
1.4.3 Extension to cartesian grids in two space dimensions . . . . .	69
1.4.4 Numerical simulation of 2D homogeneous turbulence . . . . .	72
1.4.5 Extension to moving grids: application to fluid-structure interaction .	74
1.5 Energy conservative approaches for the shallow water equations . . . . .	80
1.5.1 A well-balanced and energy conservative DG scheme on Legendre-Gauss nodes . . . . .	82
1.5.2 Well-balancedness of the energy conservative MaMEC scheme . . . . .	89
<b>2 Viscous Flow</b>	
<i>Discretizing Diffusion Terms in DG Framework</i>	<b>93</b>
2.1 DG discretization of linear diffusion in one space dimension . . . . .	95
2.2 The $(\sigma, \mu)$ -family of DG diffusion discretizations . . . . .	99
2.2.1 Connection to contemporary DG diffusion discretizations . . . . .	100

2.2.2	An energy estimate for the global diffusion operator . . . . .	102
2.3	Energy stability of the BR2 scheme . . . . .	105
2.4	A comparative Fourier analysis for linear advection-diffusion problems . . . . .	109
2.4.1	The DG-discretized linear advection-diffusion equation . . . . .	112
2.4.2	Eigensolution analysis . . . . .	113
2.4.3	Numerical results for the linear advection-diffusion equation . . . . .	127
<b>3</b>	<b>Recent Advances on Time Discretization</b>	
	<i>IMEX Advection-Diffusion Splitting and Positivity Preservation</i>	<b>137</b>
3.1	Implicit-Explicit (IMEX) approaches for fluid flow equations . . . . .	139
3.2	IMEX-RK schemes for advection-diffusion splitting . . . . .	141
3.3	$L^2$ -stability analysis of IMEX- $(\sigma, \mu)$ DG schemes for advection-diffusion . . . . .	147
3.3.1	Stability analysis with respect to the first order IMEX scheme . . . . .	150
3.3.2	Stability analysis with respect to the second order IMEX scheme . . . . .	153
3.3.3	Numerical results . . . . .	160
3.4	Positivity preservation . . . . .	175
3.4.1	The Patankar approach applied to production-destruction equations . . . . .	178
3.4.2	Quasi-linear production-destruction equations arising from PDEs . . . . .	181
3.4.3	Applying the Patankar trick to an implicit RK scheme . . . . .	191
<b>4</b>	<b>Wetting and Drying Shallow Water Flows</b>	
	<i>Designing Positivity Preserving, Conservative and Well-balanced Schemes</i>	<b>201</b>
4.1	The governing equations of shallow water flow . . . . .	202
4.2	Numerical challenges of shallow water flow simulation . . . . .	205
4.3	Review of wetting and drying procedures for FV and DG methods . . . . .	209
4.3.1	Finite Volume Methods . . . . .	209
4.3.2	Discontinuous Galerkin Schemes . . . . .	213
4.4	Wetting and drying treatment based on the Patankar trick . . . . .	221
4.4.1	Production-destruction splitting of the DG-discretized SW equations . . . . .	222
4.4.2	MPSDIRK3 time integration for the DG-discretized SW equations . . . . .	224
4.4.3	Shock capturing by modal filtering . . . . .	227
4.4.4	Numerical experiments . . . . .	229
	<b>Summary and Future Prospects</b>	<b>239</b>
	<b>Bibliography</b>	<b>243</b>

# Introduction

The current demands regarding the numerical simulation of fluid flow often require highly accurate computations to obtain a detailed resolution of the occurring physical phenomena. The basic concept for the construction of a fluid solver is to transfer the physical model into a numerical scheme which complies with the underlying physical principles such as conservation and balances of certain quantities. In addition, the numerical methods are required to be stable and efficient. Again, stability is often determined by physically motivated quantities such as energy or entropy and it is generally easier to be achieved for low order schemes. Furthermore, the need for efficiency and possible implementation in parallel hardware environment has led to the development of sophisticated schemes in space with compact stencils such as discontinuous Galerkin (DG) methods and flux reconstruction schemes which extend classical space discretization methods.

The summation-by-parts (SBP) property is a mimetic property transferring the continuous integration-by-parts rule to the discrete setting. Furthermore, it is a highly desired property in high order space discretization schemes which facilitates proofs of energy stability with respect to the interior of the computational domain and may be combined with the simultaneous-approximation-term (SAT) approach for a stable treatment of boundary conditions. In this work, both local and global SBP properties will be detected in specific classes of DG schemes and flux reconstruction methods. The potential lack of an SBP property may also be used to discard specific seemingly efficient and formally high order schemes such as early spectral difference methods in favor of provably stable methods such as energy stable flux reconstruction schemes. However, the greatest benefit to date of high order SBP schemes is their ability to discretize skew-symmetric forms of conservation laws in a manner which both satisfies the primary conservation principles and specific secondary balances comprised in the derivation of the skew-symmetric form. Originally, only classical SBP schemes have been used in this context which requires including the cell boundaries in the nodal set and excludes classical Legendre-Gauss collocation DG schemes from the realm of possible space discretizations. This is in contrast to DG schemes on Legendre-Gauss-Lobatto nodes which include a sufficient amount of integration points on cell boundaries and may be understood as classical SBP schemes on the level of elements.

In this work, the newly found generalized SBP properties of DG schemes on Legendre-Gauss nodes pave the way to their application to skew-symmetric forms. Since their quadrature rule possesses a higher degree of exactness, DG schemes on Legendre-Gauss nodes are usually more accurate than those on Legendre-Gauss-Lobatto nodes and might be preferable for long-time simulations, which is precisely the situation in which the preservation of secondary quantities

should be most beneficial. In this context, Chapter 1 reviews the SBP properties of various classes of methods and presents the construction of kinetic energy preserving Legendre-Gauss DG schemes for the Euler- and Navier-Stokes equations. The potential of these schemes regarding higher efficiency is indicated for a prototype long-time viscous flow simulation in one space dimension. Furthermore, their viability is demonstrated for simulations of decaying two-dimensional homogeneous isotropic turbulence and regarding their utilization as fluid solvers within a coupled scheme applied to a classical one-dimensional test case of mechanical fluid-structure interaction. Considering a second field of application, we construct energy conservative Legendre-Gauss DG schemes for shallow water flow over non-flat bottom topography and investigate the related discrete preservation of equilibrium states of the original balance law.

Discontinuous Galerkin and flux reconstruction schemes commonly produce piecewise polynomial approximations with discontinuities across cell interfaces which are ideally adapted to the numerical simulation of hyperbolic conservation laws. These partial differential equations admit discontinuities which may develop in finite time even though the initial conditions are smooth. In fact, the original introduction of DG schemes to these problems was viewed and promoted as a natural extension and alternative to the widely used finite volume approaches by incorporating the methodology of numerical fluxes into a higher order approximation. However, the inherent discontinuity of the DG approximate solution does not offer any intrinsic way to discretize diffusion operators. These circumstances have led to the development of a multitude of DG diffusion discretizations. Generally, these methods either use specially designed penalty terms within a finite element approach or rewrite the underlying partial differential equations of convection-diffusion type into a system of first order equations using auxiliary variables for the solution derivatives. The latter approach yields contemporary discretizations of viscous terms in DG framework such as the local discontinuous Galerkin (LDG) and Bassi-Rebay (BR) schemes. Related to the early penalty methods, namely the interior penalty and the Baumann-Oden scheme, the so-called  $(\sigma, \mu)$ -family of DG diffusion discretizations has been revisited more recently in the context of recovery-based approaches aimed at recovering a smooth approximation to increase the accuracy of the DG scheme for diffusion problems.

In this thesis, specific aspects regarding the discretization of diffusion terms in the DG framework are dealt with in Chapter 2. First, for the one-dimensional linear diffusion equation, the generalized upwind SBP properties of the LDG and BR schemes are investigated. Furthermore, we prove certain properties of the  $(\sigma, \mu)$ -family regarding symmetry and dissipativity under conditions on the two parameters and show that the BR schemes incorporated into a DG discretization on Legendre-Gauss-Lobatto nodes are members within this family. The second BR scheme termed BR2 includes a penalty parameter determining stability and accuracy of the scheme and is based on the construction of a lifting operator applied to jumps at cell interfaces. For the DG scheme on Legendre-Gauss-Lobatto nodes, the BR2 lifting operator may be calculated either by exact projection or using inexact numerical integration on the given nodes. For both versions, we determine the stability properties with respect to the penalty parameter by extending recent results on the equivalence of the BR2 scheme and the classical interior penalty formulation in one space dimension.

Regarding the use of high order schemes in space, the investigation of their wave propagation

properties in terms of dispersion and diffusion errors depending on the wave number is of utmost importance for the accuracy and stability analysis of fluid flow simulations. For these schemes and this type of application, a desired small numerical dissipation generally competes with robustness and thus has to be carefully analyzed. In the literature, dispersion and diffusion properties have thus been investigated for major classes of high order schemes applied to advection problems, usually by means of Fourier analysis. Studies of the dissipation and dispersion properties of DG schemes applied to advection-diffusion problems are more recent and also a topic of this work. Specifically, in the last section of Chapter 2, we study the influence on dissipation and dispersion properties of the two most frequently used alternating versions of the LDG scheme as well as the BR schemes. The analysis highlights a significant difference between the two possible ways to choose the alternating LDG fluxes. Furthermore, we compute a combined error measure quantifying the accuracy with respect to the wave number for a larger range of polynomial degrees to detect odd-even phenomena particularly in relation to the BR schemes and to investigate differences with respect to the specific choice of DG nodal set.

Naturally, current research regarding accurate, stable and efficient numerical schemes for fluid flow simulations also includes the aspect of time integration. Discretization in space by discontinuous Galerkin or flux reconstruction schemes results in a system of ordinary differential equations (ODEs). In principle, this allows for the straightforward application of standard ODE solvers such as Runge-Kutta (RK) or linear multistep methods to this semi-discrete system. However, advanced time integration schemes often consist in combinations of known methods or modify existing schemes, for instance to increase efficiency or stability or to transfer additional properties of the analytical solution to its discrete approximation.

A particular case of combination is merging implicit and explicit time integration into so-called IMEX schemes. While explicit schemes are comparatively easy to implement, also in parallel hardware environment, have a low computational cost per time step, and may be constructed with high order of accuracy, their range of stability is limited. However, implicit schemes applied to DG discretized fluid equations usually involve the solution of large nonlinear equations for each time step. Thus, the computational effort is larger and parallel computing is more difficult to realize. Nonetheless, the additional effort of implicit schemes can pay off if the problem is stiff, i.e. if the time step constraints ensuring stability are much more restrictive than those achieving the desired accuracy of the numerical solution. Ideally, IMEX schemes suitably adjust the favorable properties of both classes of methods to the given problem by implicitly discretizing the parts causing stiffness and treating the other parts explicitly. Considering the numerical simulation of fluid flow based on the compressible Navier-Stokes equations or related problems of advection-diffusion type, numerical stiffness may be caused for example by boundary layers and the resulting locally refined grids, by acoustic waves, or due to the presence of viscous terms. A natural choice based on stiffness due to viscous terms is advection-diffusion splitting, whereby the advective terms are discretized explicitly while the diffusive terms are treated implicitly. While advection-diffusion IMEX splitting alleviates the severe time step scaling for the diffusion terms, a stability based time step restriction of Courant-Friedrichs-Lowy (CFL) type may still have to be fulfilled due to the explicit discretization of advection terms demanding a step size reduction on locally refined grids. However, for specific combinations of space and time discretization, the IMEX stability properties are even more favorable and allow for a grid-independent time step choice. This

means that the combined IMEX scheme has better stability properties for advection-diffusion in comparison to its explicit part applied to the pure advection problem. The first part of Chapter 3 covers specifically designed IMEX-DG schemes which profit from this stabilizing effect and allow for higher efficiency due to larger time steps. While previous investigations considered IMEX advection-diffusion splitting for finite difference or Fourier-type spatial discretizations as well as for the DG scheme with LDG diffusion fluxes, the question arises if this prominent feature is also inherent to more general DG diffusion schemes. In this work, we therefore analyze the corresponding stability properties of the DG scheme using the  $(\sigma, \mu)$ -family for diffusion discretizations. This family naturally includes the original DG diffusion discretizations as well as the BR schemes and a symmetrized form of the LDG scheme, and is thus quite versatile. From a theoretical analysis, conditions on the given parameters  $\sigma$  and  $\mu$  are derived which guarantee  $L^2$ -stability for time steps  $\Delta t = \mathcal{O}(d/a^2)$ , where  $a$  and  $d$  denote the advection and diffusion coefficient, respectively. This signifies that the allowable time step size does not decrease under grid refinement.

A further aspect of time integration is positivity preservation. In fact, enforcing positivity of specific quantities involved in the description of time-dependent physical processes often requires a careful choice of the time integration scheme or the design of suitable modifications. For instance, the density and pressure in the description of compressible fluid flow should both be positive quantities. Regarding the flow of water in rivers, lakes or oceans, the water column height should be non-negative. This is a constraint which also holds for the concentrations of substances in chemical processes. In general, the approximations given by a numerical method do not necessarily satisfy these bounds. However, a violation may cause a blow-up of the numerical algorithm as the involved mathematical operations may not be well-defined anymore if the relevant quantities are located outside of the physically meaningful range. In this regard, the second part of Chapter 3 deals with positivity preserving schemes for semi-discretizations of partial differential equations in production-destruction form. In particular, we focus on the Patankar approach modifying Runge-Kutta schemes to achieve both unconditional positivity and conservation for these equations.

The last part of this thesis focuses on the application of positivity preserving DG schemes to shallow water flows including wetting and drying processes and thus links the Patankar schemes discussed in the previous chapter to a practically relevant situation. The consideration of alternating wetting and drying shallow water flows pertains in particular to the simulation of flows in rivers, lakes or coastal regions where the water depth may drastically change. Capturing the fluid dynamics for instance in coastal areas is significant both for coastal engineering and with respect to marine ecosystems. The importance of an accurate simulation of wetting and drying for diverse examples of shallow water flows is accompanied by challenges with respect to the construction of accurate, robust and efficient numerical methods.

In this context, Chapter 4 reviews a multitude of numerical methods for wetting and drying shallow water flows in two space dimensions using fixed grids and applying finite volume or discontinuous Galerkin space discretizations. Hereby, the major challenges faced by numerical schemes tackling the simulation of wetting and drying processes are discussed in more detail. One of these challenges is the preservation of a non-negative water column height requiring the use of positivity preserving time integrators. Positivity preserving explicit time integration

may be built upon particular Runge-Kutta schemes which can be written as convex combinations of explicit forward Euler steps. In view of their monotonicity properties with respect to convex functions, these time integration methods are called strong-stability preserving (SSP) schemes. Combined with positivity preserving numerical fluxes, explicit SSP-RK schemes play a leading role in the construction of positivity preserving finite volume and DG schemes. Regarding the literature on numerical methods for wetting and drying shallow water flows, explicit time stepping is indeed implemented in the majority of suggested algorithms. Positivity preserving explicit schemes generally provide physically sound results for this application as they can accurately represent the dynamics of the flooding and receding front due to the necessarily small time step sizes. Unfortunately, there are limits to the efficiency of explicit time integration schemes, especially if the mesh is refined in the shallow regions of the flow. For instance, large local variations of the bottom topography may lead to this kind of grid induced stiffness. In this case, implicit schemes can yield significant speed-up and permit simulations that would be impossible to carry out with explicit methods. However, positivity preservation by implicit SSP-RK schemes enforces non-negative cell means of water height under rather restrictive time step constraints that depend on the cell sizes and the order of the DG discretization. Since these positivity enforced time step restrictions interfere with the efficiency of the implicit time integrator, unconditionally positive implicit schemes are desired. However, it is known that any general linear method that is unconditionally positivity preserving can only be first order accurate at best. The Patankar schemes considered in this work avoid this order restriction since they do not belong to the class of general linear methods. In order to employ them for DG discretized shallow water equations, a production-destruction formulation is first extracted from the semi-discrete continuity equation which describes the time evolution of the water height averages within a DG scheme on triangular grids. The strategy of positivity preservation by explicit SSP-RK schemes is then extended to implicit time integration by the Patankar approach. The weights on the destruction terms which are introduced by the Patankar scheme are thereby designed to reduce the outgoing water fluxes while applying corresponding weights to the production terms guarantees mass conservation. Due to the use of an implicit Runge-Kutta scheme as base method for the Patankar approach, the resulting scheme can now take full advantage of larger time steps and is therefore able to beat explicit time stepping in terms of the required computational time.



# Chapter 1

## High Order Schemes for Conservation Laws in Fluid Dynamics

### *Benefit of SBP Properties in Modern Space Discretization Schemes*

The time dependent partial differential equations (PDEs) describing fluid flow are commonly discretized using the so-called *method of lines*. This approach first discretizes the spatial derivatives and thus yields a system of ordinary differential equations (ODEs) of a size and structure depending on the specifically chosen space discretization scheme.

Modern space discretization schemes for fluid flow are still closely related to the classical discretization techniques of *finite difference (FD)*, *finite volume (FV)* and *finite element (FE)* methods depicted and characterized in Table 1.1. These general approaches can be distinguished by the specific form of the continuous fluid equations they discretize. These fluid equations are obtained from the physical principles of mass, momentum and energy conservation which leads to the integral form

$$\frac{d}{dt} \int_V \mathbf{u} d\mathbf{x} + \int_{\partial V} \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) \cdot \mathbf{n} d\sigma = \int_V \mathbf{s}(\mathbf{u}, \mathbf{x}, t) d\mathbf{x},$$

where  $\mathbf{u} : (\mathbf{x}, t) \rightarrow \mathbb{R}^m$  contains the conserved variables,  $\mathbf{F}$  is a flux tensor,  $\mathbf{s}$  is a source term,  $V$  is a control volume which is fixed in space and  $\mathbf{n}$  is the outward pointing normal vector at  $\partial V$ . This *integral form*, which is directly based on the conservation properties, is the starting point for the finite volume scheme using the cell means  $\bar{\mathbf{u}}_V = \frac{1}{|V|} \int_V \mathbf{u} d\mathbf{x}$  as degrees of freedom. From the integral form we may obtain a corresponding partial differential equation of the form

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{s}$$

by using the Gauss divergence theorem. This is the *differential form* of the given conservation laws. This formulation is generally used by finite difference schemes which use pointwise

(*nodal*) values as degrees of freedom and approximate the spatial derivatives by differences of these nodal values. From the differential form of conservation laws, a so-called *weak form* may be constructed by multiplying the differential form with *test functions* and integrating over the whole domain. This weak form is then used by finite element methods which specify a finite number of test functions and represent the approximate solution as an expansion in a finite set of space-dependent *basis functions*, also called *ansatz functions*. The resulting equations are also referred to as the *weighted residual formulation*. As listed in Table 1.1, although the advantage of finite difference approximations is the simple method structure and high efficiency on structured grids, they are much more difficult to generalize to unstructured grids in two and three space dimensions.

Finite volume approximations possess one degree of freedom per conserved variable and computational grid cell, i.e. the cell mean, and the time evolution of these cell means is given by fluxes through cell boundaries. Hence this procedure easily generalizes to arbitrary decompositions of the spatial domain into a computational grid. While finite difference schemes need a specific structure to comply with conservation properties, finite volume schemes are conservative by construction. However, unlike finite difference and finite element methods, a space discretization of higher order is more difficult to obtain with finite volume schemes as such high order extensions are based on a suitable interpolation of adjacent cell means.

Finite element methods share the advantage of flexibility in terms of the computational grid with finite volume schemes. Higher order in space is more easily obtained by a larger set of basis functions on a given grid cell, which may also be regarded as using more interior nodes within each cell. More modern space discretizations share approaches of more than one of these classes. For example, the *discontinuous Galerkin (DG)* scheme may be derived by a finite element approach using discontinuous basis and test functions. Therefore, finite element techniques play a substantial role in the analysis of DG schemes. Furthermore, by construction, the DG approximate solution is generally discontinuous over element boundaries. In order to deal with these discontinuities, the concept of *numerical flux functions* inherent to finite volume methods is incorporated into the DG scheme. Finally, nodal DG schemes using Lagrange polynomials to a certain set of nodes as basis and test functions may be rewritten both as SBP schemes with similarities to finite difference methods and as subcell finite volume schemes. For brevity, the major difference between the finite volume and the discontinuous Galerkin approaches consists in the construction of the desired higher order discretizations. While finite volume methods rely on reconstruction of pointwise data using stencils of adjacent cells to achieve higher order in space, for the discontinuous Galerkin approximation, the polynomial degree of the test and basis functions is simply increased.

After space-discretization, one of the first steps within the theoretical analysis of a proposed method then consists in the assessment of the stability and accuracy of this system of ODEs called the *semi-discrete scheme* under spatial refinement. At best, the semi-discrete scheme is constructed to fulfill certain properties which ensure that the solutions to the semi-discrete equations mimic the behavior of the solution to the underlying PDE. The so-called *summation-by-parts* property falls into this construction category as it mimics the integration-by-parts formula in a discrete sense. With this property, energy stability may be ensured for a variety of PDEs. In particular, SBP schemes guarantee stability for the linear advection equation, they are thus linearly stable. It is worth to mention that some early spectral difference

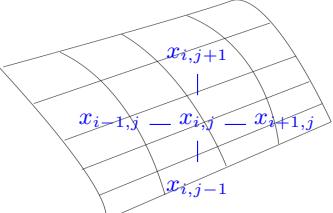
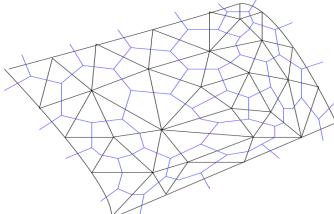
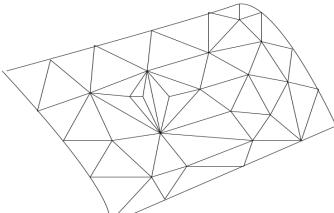
<p style="text-align: center;">Finite differences (FD): based on the differential form</p>	
<ul style="list-style-type: none"> <li>• approximation of nodal values and nodal derivatives</li> <li>• easy to derive, efficient</li> <li>• essentially limited to structured meshes</li> </ul>	
<p style="text-align: center;">Finite volumes (FV): based on the integral form</p>	
<ul style="list-style-type: none"> <li>• approximation of cell means and integrals</li> <li>• conservative by construction</li> <li>• suitable for arbitrary meshes</li> <li>• difficult to extend to higher order</li> </ul>	
<p style="text-align: center;">Finite elements (FE): based on the weak form</p>	
<ul style="list-style-type: none"> <li>• weighted residual formulation</li> <li>• quite flexible and general</li> <li>• suitable for arbitrary meshes</li> </ul>	

Table 1.1: General space discretization schemes.

methods which are members of the class of flux reconstruction schemes did not possess an SBP property and hence suffered from instability for the linear advection equation. This lack of structural property has been remedied by the class of energy stable flux reconstruction (ESFR) schemes [202, 36] discussed in Section 1.3.

### Organization of this chapter

In this chapter, we will first review the general framework of SBP operators also providing the historical background regarding the SBP property of finite difference schemes. For this purpose, the following Section 1.1 takes into account both classical and generalized SBP operators for the discretization of derivatives of first and second order and also discusses the new class of upwind SBP schemes. In this regard, we highlight the favorable properties of SBP schemes, in particular their fulfillment of discrete energy estimates, their connections to mimetic schemes via a discrete counterpart to the integration-by-parts rule, and the facilitation of conservative discretizations of conservation laws in skew-symmetric form.

Subsequently, the focus of Section 1.2 is on the detection of SBP properties within nodal discontinuous Galerkin schemes, also including the case that their nodal set in one space dimension does not include the cell boundary points. For instance, the DG scheme on Legendre-Gauss nodes falls into this category. On the one hand, the SBP properties of the DG scheme in one-space dimension, on tensor-product grids and on unstructured triangular grids are derived locally on the level of elements where cell interfaces are dealt with the boundary operators  $\mathbf{B}$  of generalized SBP schemes. In this context, the familiar subcell finite volume property of the DG scheme on Legendre-Gauss-Lobatto nodes is extended to the DG scheme using the more accurate Legendre-Gauss nodes. On the other hand, on a global level, specific DG schemes may be considered as generalized upwind SBP schemes where the cell interface terms, together with interior contributions, are globally assembled into a discrete derivative operator of upwind SBP type.

The class of flux reconstruction schemes generalizes various high order space discretization methods for computational fluid dynamics via their differential formulation. Acting on the observation that some early schemes in this class suffer from a weak instability for linear problems, the subclass of ESFR schemes has been identified in the literature as mentioned above. Section 1.3 investigates the close link between SBP properties and ESFR schemes and shows that all known ESFR schemes, either in one space dimension or on unstructured triangular grids are in fact generalized SBP schemes in the local sense considering the formulation on each element.

The last two sections of this chapter are dedicated to the derivation of space discretization methods additionally enforcing the discrete balance of specific secondary quantities which may become relevant in fluid flow simulations. The basic idea enabling these additional properties is to utilize skew-symmetric formulations of the underlying conservation laws. In Sections 1.4 and 1.5, we thereby transfer results from the key papers [61, 68] exploiting the more classical SBP properties of DG schemes on Legendre-Gauss-Lobatto nodes to DG schemes on Legendre-Gauss nodes obeying a generalized SBP property. Section 1.4 thus constructs a kinetic energy preserving DG scheme on Legendre-Gauss nodes for the Euler- and Navier-Stokes equations. Numerically, the expected higher accuracy resulting from the higher degree of exactness of

the Legendre-Gauss quadrature rule is then demonstrated for a prototype long-time viscous flow simulation in one space dimension. Further simulations are carried out for decaying two-dimensional homogeneous isotropic turbulence and for a classical one-dimensional test case of mechanical fluid-structure interaction. Finally, in Section 1.5, the Legendre-Gauss DG scheme is applied to a skew-symmetric formulation of the shallow water equations with non-flat bottom topography leading to the preservation of total energy which represents an entropy function of this system of conservation laws. Focus of that section is on the relation of this property to the preservation of moving water equilibria which generalize the lake at rest steady state solution. In this regard, we compare the skew-symmetric energy conservative DG schemes to a staggered scheme conserving mass, momentum and energy developed in [79].

## 1.1 The SBP property: Historical background and generalizations

The summation-by-parts (SBP) property is a mimetic property – mimicking the analytical concept of integration-by-parts. Originally, discrete derivative operators with an SBP property have been introduced to finite difference schemes in combination with suitable boundary closures by Kreiss and Scherer [104]. The SBP concept for finite difference schemes has been revisited and further developed by Strand [182], Carpenter et al. [32] and Olsson [146, 147] in order to construct high order accurate, conservative and stable numerical methods. More precisely, via  $L^2$  energy estimates, the spatial SBP operators automatically yield stable schemes for periodic solutions to the linear advection-diffusion equation in one space dimension. In addition, they have substantially profited from a combination with weakly enforced boundary conditions, most prominently by using *simultaneous approximation terms (SATs)* which were first developed in [32] and later extended to an interface treatment in case of multiple domains in [33]. This combination allowed for stability proofs for more complicated problems and systems of PDEs including the linearized compressible Navier-Stokes equations dealt with in [144]. Reviews on SBP operators and SAT terms are given by Svärd and Nordström [187] and Del Rey Fernández et al. [48] showing that the methodology enables the construction of stable schemes for quite general cases also including variable coefficient equations and nonlinear conservation laws like Burgers' equation.

At present, the theory and application of summation-by-parts (SBP) operators is still an active field of research for the numerical solution of time-dependent partial differential equations. Moreover, not only the original background of finite difference schemes is considered as recent results also focus on SBP operators within various popular classes of numerical schemes, e.g. finite volume schemes on unstructured dual grids [143], discontinuous Galerkin(DG) schemes with Legendre-Gauss-Lobatto nodes [62], the correction-via-reconstruction scheme which includes so-called spectral-difference methods, see [166], as well as the direct construction of generalized nodal SBP operators in one space dimension [47] and even on simplex elements [77]. Based on the generalized SBP property in [47], DG schemes on Legendre-Gauss nodes both in one space dimension and on tensor-product grids have been identified as generalized SBP schemes in [150]. In fact, the definition in [47] allows to derive generalized SBP properties for 1D nodal DG schemes on arbitrary nodal sets, as long as their corresponding quadrature rule is sufficiently exact in order to mimic integration-by-parts at the discrete level.

The properties of classical SBP schemes can best be illustrated regarding their original background of linear or linearized PDEs based on first derivative operators, e.g. considering the 1D linear advection equation

$$\frac{\partial}{\partial t}u(x,t) + a\frac{\partial}{\partial x}u(x,t) = 0, \quad \alpha \leq x \leq \beta, \quad t > 0. \quad (1.1)$$

The constant advection velocity  $a$  is assumed to be positive,  $a > 0$ , and for simplicity of the subsequent analysis, a homogeneous boundary condition  $u(\alpha, t) = 0$  are posed. Exact solutions to this problem admit an estimate for the behavior in time of the  $L^2$  energy defined by

$$\|u(x,t)\|_{L^2(\alpha,\beta)}^2 = \int_{\alpha}^{\beta} u^2(x,t) dx.$$

This estimate is derived from (1.1) using the energy method, i.e. (1.1) is multiplied by  $u(x,t)$  and integrated in space,

$$\int_{\alpha}^{\beta} u(x,t) \frac{\partial}{\partial t}u(x,t) dx + a \int_{\alpha}^{\beta} u(x,t) \frac{\partial}{\partial x}u(x,t) dx = 0.$$

Using  $u \frac{\partial}{\partial x}u = \frac{1}{2} \frac{\partial}{\partial x}u^2$  and the boundary condition  $u(\alpha, t) = 0$ , we then have

$$\frac{d}{dt} \|u(x,t)\|_{L^2(\alpha,\beta)}^2 + au^2(\beta,t) = 0. \quad (1.2)$$

SBP schemes desire to transfer this estimate obtained for the continuous problem to the discrete or semi-discrete case.

Classical SBP finite difference operators approximating the first derivative  $\frac{\partial}{\partial x}$  are given on a subdivision of the domain  $[\alpha, \beta]$  by equidistant grid points which include the domain boundaries. Denoting these grid points by  $x_j = \alpha + j\Delta x$ ,  $j = 0, \dots, N$ , with mesh width  $\Delta x = \frac{\beta - \alpha}{N}$ , we define an approximate solution to (1.1) by the time-dependent solution vector  $\mathbf{u}(t)$  given as

$$\mathbf{u}(t) = (u_0(t), \dots, u_N(t))^T \approx (u(x_0, t), \dots, u(x_N, t))^T.$$

A classical SBP finite difference scheme to solve (1.1) is of the form

$$\frac{d\mathbf{u}}{dt} + a\mathbf{D}\mathbf{u} = \sigma\mathbf{M}^{-1}\mathbf{e}_0 u_0, \quad \mathbf{e}_0 = (1, 0, \dots, 0)^T, \quad (1.3)$$

where the matrix  $\mathbf{D}$  is a first-derivative SBP operator with the following properties and  $\sigma \in \mathbb{R}$  is a parameter which has yet to be specified.

**Definition 1.1.** *The finite difference scheme (1.3) is an SBP scheme with first-derivative SBP operator  $\mathbf{D}$  of order and degree  $q$  if*

1. *The matrix  $\mathbf{D}$  is an accurate approximation to  $\frac{\partial}{\partial x}$  with*

$$\mathbf{D}\mathbf{x}^k = k\mathbf{x}^{k-1}, \quad 0 \leq k \leq q,$$

*where  $\mathbf{x}^k = (x_0^k, \dots, x_N^k)^T$  is the representation of the monomials  $x^k$  on the grid points.*

2. The matrix  $\mathbf{M}$  is symmetric and positive definite.
3. Setting  $\mathbf{S} = \mathbf{M}\mathbf{D}$ , integration by parts is mimicked by the property

$$\mathbf{S} + \mathbf{S}^T = \mathbf{M}\mathbf{D} + \mathbf{D}^T\mathbf{M} = \mathbf{B}, \quad (1.4)$$

with  $\mathbf{B} = \text{diag}(-1, 0, \dots, 0, 1)$ .

The matrix  $\mathbf{M}$  is also referred to as the *norm* of the SBP operator  $\mathbf{D}$  since due to its symmetry and positive definiteness, an inner product and a norm may be defined by  $(\mathbf{u}, \mathbf{v})_{\mathbf{M}} = \mathbf{u}^T \mathbf{M} \mathbf{v}$  and  $\|\mathbf{u}\|_{\mathbf{M}} = \sqrt{\mathbf{u}^T \mathbf{M} \mathbf{u}}$ . Furthermore, the right-hand side term of the equation (1.3) containing the parameter  $\sigma$  is an example of an SAT term which weakly enforces the homogeneous boundary condition.

At this point, we remark that the *order* of a derivative operator refers to the leading truncation error term, i.e. the first-derivative operator  $\mathbf{D}$  is of order  $q$  if and only if

$$(\mathbf{D} \mathbf{v})_j = \frac{dv}{dx}(x_j) + \mathcal{O}((\Delta x)^q), \quad j = 0, \dots, N, \quad (1.5)$$

for a sufficiently smooth function  $v : [\alpha, \beta] \rightarrow \mathbb{R}$  and its grid representation

$$\mathbf{v} = (v(x_0), \dots, v(x_N))^T.$$

On the other hand, the *degree* of a derivative operator is the highest degree of a monomial, represented by  $\mathbf{x}^k$ , for which the operator is exact. For an operator approximating the  $m$ th derivative, we have the relation

$$\text{order} = \text{degree} - m + 1. \quad (1.6)$$

One may derive the relation (1.6) for difference operators using finite stencils by Taylor expansion of the smooth function  $v$ , which yields

$$v(x) = T_k(x) + \frac{v^{(k+1)}(\xi)}{k!}(x - x_j)^{k+1},$$

where  $T_k(x)$  is the  $k$ th degree Taylor polynomial of  $v$  at the point  $x_j$ , given by

$$T_k(x) = v(x_j) + v'(x_j)(x - x_j) + \dots + \frac{v^{(k)}(x_j)}{k!}(x - x_j)^k,$$

and  $\xi$  is some real number between  $x$  and  $x_j$ . Hereby, the index  $j$  corresponds to  $j$  in (1.5). For an operator  $\mathbf{D}_m$  of degree  $q$  approximating the  $m$ th derivative, with  $m \leq q$ , we have

$$(\mathbf{D}_m \mathbf{v})_j = (\mathbf{D}_m \underline{T}_q)_j + \frac{v^{(q+1)}(\xi)}{q!}(\Delta x)^{1-q}(\mathbf{D}_m \mathbf{r}^{q+1})_j = v^{(m)}(x_j) + \mathcal{O}((\Delta x)^{q+1-m})$$

where the vectors  $\underline{T}_q$  and  $\mathbf{r}$  are defined by  $\underline{T}_q = (T_q(x_0), \dots, T_q(x_N))^T$  and  $\mathbf{r} = (r_0, \dots, r_n)$  with components  $r_k = (k - j)$ , respectively, and we used the fact that the entries of an  $m$ th derivative operator are of order  $\mathcal{O}((\Delta x)^m)$ .

Hence, for the first-derivative, the order of the discrete derivative operator is equal to its degree.

### A discrete energy estimate

For the analysis of the SBP scheme (1.3), the behavior of the discrete quantity  $\|\mathbf{u}\|_{\mathbf{M}}^2$ , which may be seen as an approximation of the  $L^2$  energy  $\|u\|_{L^2}^2$ , is investigated by the energy method similar to the continuous case. Multiplying (1.3) from the left by  $\mathbf{u}^T \mathbf{M}$  yields

$$\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt} + a \mathbf{u}^T \mathbf{M} \mathbf{D} \mathbf{u} = \sigma u_0^2. \quad (1.7)$$

Similarly, multiplying the transpose of (1.3) from the right by  $\mathbf{M} \mathbf{u}$  while considering the symmetry of  $\mathbf{M}$ , results in

$$\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt} + a \mathbf{u}^T \mathbf{D}^T \mathbf{M} \mathbf{u} = \sigma u_0^2. \quad (1.8)$$

Adding (1.7) and (1.8) now yields

$$2\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt} + a \mathbf{u}^T (\mathbf{M} \mathbf{D} + \mathbf{D}^T \mathbf{M}) \mathbf{u} = 2\sigma u_0^2. \quad (1.9)$$

Using  $\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 = 2\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt}$  and the SBP property (1.4), we arrive at

$$\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 + a(u_N^2 - u_0^2) = 2\sigma u_0^2$$

For  $\sigma \leq -\frac{a}{2}$ , the time evolution of the discrete energy may therefore be estimated by

$$\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 + au_N^2 \leq 0, \quad (1.10)$$

resulting in a stable numerical scheme. For  $\sigma = -\frac{a}{2}$ , we have equality in (1.10) and the bound on the discrete energy in fact directly corresponds to the continuous case (1.2).

### Connection to mimetic schemes

SBP schemes fall into the category of mimetic schemes which have lately been reviewed in [115]. In fact, mimetic schemes have a long history of more than 50 years of development. The idea of mimetic discretizations is to derive discrete analogs of the differential operators occurring in PDEs – such as gradient, divergence and curl – which exactly mimic the mathematical properties of the corresponding continuous versions. Hereby, the underlying PDEs are usually reformulated by using first-order operators and properties of higher order operators follow from discrete duality relations. Based on a set of primary operators, a discrete vector and tensor calculus is then developed using mimetic design principles.

### Discretization of the integration-by-parts rule

The gist of the SBP property (1.4) is that it provides a discrete counterpart of the continuous rule of integration by parts. More precisely, for two functions  $u, v : [\alpha, \beta] \rightarrow \mathbb{R}$  with grid

representations  $\mathbf{u}$  and  $\mathbf{v}$ , respectively, we have

$$\begin{aligned} \int_{\alpha}^{\beta} u \frac{\partial}{\partial x} v dx &\approx \mathbf{u}^T \mathbf{M} \mathbf{D} \mathbf{v} & \stackrel{(1.4)}{=} & -\mathbf{u}^T \mathbf{D}^T \mathbf{M} \mathbf{v} + u_N v_N - u_0 v_0 \\ &\stackrel{\mathbf{M} \text{ symmetric}}{=} -\mathbf{v}^T \mathbf{M} \mathbf{D} \mathbf{u} + u_N v_N - u_0 v_0 \\ &\approx -\int_{\alpha}^{\beta} v \frac{\partial}{\partial x} u dx + [uv]_{x_0}^{x_N} \end{aligned}$$

Therefore, by (1.4), we obtain a discrete representation of the chain rule and subsequent integration, i.e. of  $\int_{\alpha}^{\beta} u \frac{\partial}{\partial x} u dx = \frac{1}{2} \int_{\alpha}^{\beta} \frac{\partial}{\partial x} u^2 dx = \frac{1}{2} (u^2(\beta, t) - u^2(\alpha, t))$  which is used to derive the continuous bound (1.2) on the  $L^2$  energy.

### The SBP property and high order quadrature

Less often it is emphasized that for a diagonal norm SBP operator of degree  $q$ , the approximation

$$\int_{\alpha}^{\beta} u \frac{\partial}{\partial x} v dx \approx \mathbf{u}^T \mathbf{M} \mathbf{D} \mathbf{v},$$

where  $\mathbf{M}$  is a diagonal matrix, is actually exact if  $u$  and  $v$  are polynomials of degree smaller or equal to  $q$ , see e.g. [77]. In fact,  $\mathbf{M}$  defines an accurate quadrature rule of degree  $2q - 1$ , as proven in [47], and  $\|\mathbf{u}\|_{\mathbf{M}}^2$  truly is a suitable discrete counterpart of  $\|u\|_{L^2}^2$ . At first sight, the accuracy of  $\mathbf{M}$  as a quadrature rule is quite surprising since only the accuracy of  $\mathbf{D}$  as a discrete derivative operator is part of the Definition 1.1 of an SBP scheme. However, the accuracy of  $\mathbf{D}$  is transferred to  $\mathbf{M}$  through the SBP property (1.4). Since the above approximation to the integral  $\int_{\alpha}^{\beta} u \frac{\partial}{\partial x} v dx$  is only as accurate as the accuracy of  $\mathbf{M}$  as a quadrature rule, this does not directly transfer to dense-norm SBP operators  $\mathbf{D} = \mathbf{M}^{-1} \mathbf{S}$ , where  $\mathbf{M}$  is not diagonal. The reason is that due to the larger number of degrees of freedom in the construction of dense-norm SBP operators, the degree  $q$  of the derivative operator  $\mathbf{D}$  may be higher than for diagonal-norm SBP operators constructed from the same quadrature rule incorporated in  $\mathbf{M}$ , see [47].

### Multidomain SBP operators

The enforcement of weak boundary conditions via the SAT method has also been transferred to an interface treatment for multidomain discretizations. The complete computational domain is hereby subdivided into a finite number of subdomains where possibly different SBP operators are applied. Consequently, suitable SAT terms at the subdomain interfaces have to be chosen which guarantee stability of the resulting scheme as well as discrete conservation.

As an example, we consider the linear advection equation (1.1), where the domain  $[\alpha, \beta]$  is subdivided into two subdomains  $[\alpha, \gamma]$  and  $[\gamma, \beta]$ , with  $\alpha < \gamma < \beta$ . On the left subdomain  $[\alpha, \gamma]$ , the approximate solution on equidistant grid nodes is denoted by  $\mathbf{u}_L$  while on the right subdomain, the approximate solution is denoted by  $\mathbf{u}_R$ . For the purpose of analyzing only the interface treatment, the left boundary condition in the left subdomain and the right boundary

condition in the right subdomain are ignored in the subsequent elaboration. The resulting SBP scheme with interface SATs is then given by the two equations

$$\begin{aligned} \frac{d\mathbf{u}_L}{dt} + a\mathbf{D}_L \mathbf{u}_L &= \sigma_L \mathbf{M}_L^{-1} \mathbf{e}_N (u_{L,N} - u_{R,0}), \\ \frac{d\mathbf{u}_R}{dt} + a\mathbf{D}_R \mathbf{u}_R &= \sigma_R \mathbf{M}_R^{-1} \mathbf{e}_0 (u_{R,0} - u_{L,N}), \end{aligned} \quad (1.11)$$

with  $\mathbf{e}_0 = (1, 0, \dots, 0)^T$  and  $\mathbf{e}_N = (0, \dots, 0, 1)^T$ .

For the analysis of the conservation property of the resulting scheme, the equations (1.11) are left-multiplied by  $\mathbf{1}_L^T \mathbf{M}_L$  and  $\mathbf{1}_R^T \mathbf{M}_R$ , respectively, where  $\mathbf{1}_L$ ,  $\mathbf{1}_R$  are vectors of the form  $\mathbf{1} = (1, \dots, 1)^T$ , with length corresponding to the number of grid points on the respective subdomain. Summing up, we obtain

$$\begin{aligned} \frac{d}{dt} \int_\alpha^\beta u dx &= \frac{d}{dt} \int_\alpha^\gamma u dx + \frac{d}{dt} \int_\gamma^\beta u dx \\ &\approx \mathbf{1}_L^T \mathbf{M}_L \frac{d\mathbf{u}_L}{dt} + \mathbf{1}_R^T \mathbf{M}_R \frac{d\mathbf{u}_R}{dt} \\ &= -a (\mathbf{1}_L^T \mathbf{M}_L \mathbf{D}_L \mathbf{u}_L + \mathbf{1}_R^T \mathbf{M}_R \mathbf{D}_R \mathbf{u}_R) + \sigma_L (u_{L,N} - u_{R,0}) + \sigma_R (u_{R,0} - u_{L,N}) \\ &\stackrel{(1.4)}{=} a \left( (\mathbf{D}_L \mathbf{1}_L)^T \mathbf{M}_L \mathbf{u}_L + (\mathbf{D}_R \mathbf{1}_R)^T \mathbf{M}_R \mathbf{u}_R + u_{L,0} - u_{L,N} + u_{R,0} - u_{R,N} \right) \\ &\quad + \sigma_L (u_{L,N} - u_{R,0}) + \sigma_R (u_{R,0} - u_{L,N}) \\ &= a (u_{L,0} - u_{L,N} + u_{R,0} - u_{R,N}) + \sigma_L (u_{L,N} - u_{R,0}) + \sigma_R (u_{R,0} - u_{L,N}), \end{aligned}$$

since  $\mathbf{D} \mathbf{1} = \mathbf{0} = (0, \dots, 0)^T$  due to the accuracy requirement for the derivative operator  $\mathbf{D}$  in Definition 1.1. Ignoring the terms containing  $u_{L,0}$  and  $u_{R,N}$  introduced by the left boundary of the left domain and the right boundary of the right domain, discrete conservation requires

$$(u_{R,0} - u_{L,N}) (a - \sigma_L + \sigma_R) = 0$$

and therefore the condition

$$\sigma_R = \sigma_L - a \quad (1.12)$$

on the SAT penalty parameters. Stability of the scheme (1.11) with penalty parameters satisfying (1.12) is determined by considering

$$\begin{aligned} \frac{d}{dt} \|u\|_{L^2(\alpha,\beta)}^2 &\approx 2 \mathbf{u}_L^T \mathbf{M}_L \frac{d\mathbf{u}_L}{dt} + 2 \mathbf{u}_R^T \mathbf{M}_R \frac{d\mathbf{u}_R}{dt} \\ &\stackrel{(1.11)}{=} -au_L^T (\mathbf{D}_L \mathbf{M}_L + \mathbf{M}_L \mathbf{D}_L^T) \mathbf{u}_L - au_R^T (\mathbf{D}_R \mathbf{M}_R + \mathbf{M}_R \mathbf{D}_R^T) \mathbf{u}_R \\ &\quad + 2\sigma_L u_{L,N} (u_{L,N} - u_{R,0}) + 2(\sigma_L - a)u_{R,0} (u_{R,0} - u_{L,N}) \\ &\stackrel{(1.4)}{=} a (u_{L,0}^2 - u_{L,N}^2 + u_{R,0}^2 - u_{R,N}^2) \\ &\quad + 2\sigma_L u_{L,N} (u_{L,N} - u_{R,0}) + 2(\sigma_L - a)u_{R,0} (u_{R,0} - u_{L,N}). \end{aligned}$$

Again, the influence of the outer boundaries is ignored and only the contributions of the interface  $x = \gamma$  are taken into account to bound the  $L^2$  energy. Hence, we ignore  $u_{L,0}$  and

$u_{R,N}$  and assume the corresponding terms to be bounded by SAT terms either for a domain boundary or for an additional interface. Therefore, stability requires

$$\begin{aligned} 0 &\geq a(u_{R,0}^2 - u_{L,N}^2) + 2\sigma_L u_{L,N} (u_{L,N} - u_{R,0}) + 2(\sigma_L - a)u_{R,0} (u_{R,0} - u_{L,N}) \\ &= (2\sigma_L - a)(u_{L,N} - u_{R,0})^2, \end{aligned}$$

which holds under the condition

$$\sigma_L \leq \frac{a}{2}. \quad (1.13)$$

The above interface treatment via the SAT method bears close resemblance to the incorporation of *numerical fluxes* which are inherent to finite volume methods. In fact, if the penalty parameters  $\sigma_L, \sigma_R$  satisfy the condition (1.12) enforcing a conservative scheme, then the SBP scheme (1.11) rewrites as

$$\begin{aligned} \frac{d\mathbf{u}_L}{dt} + a\mathbf{D}_L \mathbf{u}_L &= \mathbf{M}_L^{-1} \mathbf{e}_N (a u_{L,N} - (au)^*) , \\ \frac{d\mathbf{u}_R}{dt} + a\mathbf{D}_R \mathbf{u}_R &= -\mathbf{M}_R^{-1} \mathbf{e}_0 (a u_{R,0} - (au)^*) , \end{aligned} \quad (1.14)$$

with the parameterized numerical flux given by

$$(au)^* = \frac{a}{2}(u_{L,N} + u_{R,0}) + a\theta(u_{L,N} - u_{R,0}),$$

for  $\theta = (a - 2\sigma_L)/(2a)$ . For the parameter  $\theta$  of the numerical flux function, we therefore obtain  $\theta \geq 0$  from the stability condition (1.13). This obviously includes the central flux for  $\theta = 0$  and the upwind flux for  $\theta = \frac{1}{2}$  which both produce energy stable semi-discrete schemes.

### Second-derivative SBP operators

The goal in deriving second-derivative SBP operators is to obtain discrete energy estimates which are again mimetic of the continuous case. Taking for example the linear heat equation

$$\frac{\partial}{\partial t} u(x, t) = \frac{\partial^2}{\partial x^2} u(x, t), \quad \alpha \leq x \leq \beta, \quad t > 0, \quad (1.15)$$

the energy method leads to

$$\frac{d}{dt} \|u(x, t)\|_{L^2(\alpha, \beta)}^2 = -2 \int_{\alpha}^{\beta} \left( \frac{\partial}{\partial x} u(x, t) \right)^2 dx + 2 \left( u(\beta, t) \frac{\partial}{\partial x} u(\beta, t) - u(\alpha, t) \frac{\partial}{\partial x} u(\alpha, t) \right). \quad (1.16)$$

On the other hand, applying a first-derivative SBP operator twice yields a second-derivative SBP operator  $\mathbf{D}_2$  which can be reformulated using the SBP property (1.4) as

$$\mathbf{D}_2 := \mathbf{D} \mathbf{D} = \mathbf{M}^{-1} (-\mathbf{D}^T \mathbf{M} \mathbf{D} + \mathbf{B} \mathbf{D}), \quad (1.17)$$

with  $\mathbf{B} = \text{diag}(-1, 0, \dots, 0, 1)$ . Neglecting boundary conditions, the corresponding semi-discretization is given by

$$\frac{d\mathbf{u}}{dt} = \mathbf{D}_2 \mathbf{u} = \mathbf{M}^{-1} (-\mathbf{D}^T \mathbf{M} \mathbf{D} + \mathbf{B} \mathbf{D}) \mathbf{u}. \quad (1.18)$$

For the energy method applied to (1.18), we therefore obtain

$$\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 = 2\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt} = -2\mathbf{u}^T \mathbf{D}^T \mathbf{M} \mathbf{D} \mathbf{u} + 2\mathbf{u}^T \mathbf{B} \mathbf{D} \mathbf{u}, \quad (1.19)$$

which is mimetic of (1.16).

In order to obtain a semi-discrete energy estimate, it is still sufficient to replace  $\mathbf{D}^T \mathbf{M} \mathbf{D}$  by a symmetric, positive semi-definite matrix  $\mathbf{A}$ . Therefore, the definition of a classical finite difference second-derivative SBP operator is the following, where we recall the relation (1.6) between order and degree of the derivative operator.

**Definition 1.2.** *A second-derivative SBP operator  $\mathbf{D}_2$  has the form*

$$\mathbf{D}_2 = \mathbf{M}^{-1} (-\mathbf{A} + \mathbf{B} \mathbf{D}), \quad (1.20)$$

where  $\mathbf{A}$  is a symmetric, positive semi-definite matrix,  $\mathbf{B} = \text{diag}(-1, 0, \dots, 0, 1)$ , and  $\mathbf{D}$  is a first-derivative SBP operator with norm  $\mathbf{M}$ .

Furthermore, the SBP operator  $\mathbf{D}_2$  is of order  $q$  and degree  $q+1$  if it satisfies the degree conditions

$$\mathbf{D}_2 \mathbf{x}^k = k(k-1)\mathbf{x}^{k-2}, \quad 0 \leq k \leq q+1, \quad (1.21)$$

where  $\mathbf{x}^k = (x_0^k, \dots, x_N^k)^T$ .

If the first-derivative operator is applied twice, i.e.  $\mathbf{D}_2 = \mathbf{D} \mathbf{D}$  for an SBP operator  $\mathbf{D}$  of order and degree  $q$ , then the conditions (1.21) are only fulfilled for  $k \leq q$  and the order of the second-derivative operator is lower by one compared to the first-derivative operator. In addition, for classical finite difference SBP operators with repeated interior stencils the bandwidth is nearly doubled and Fourier analysis shows a lack of damping for the highest frequency mode as shown in [127]. Due to these drawbacks, narrow stencil and order-matched second-derivative operators have been constructed in [128], which satisfy the SBP property (1.20).

### Discretization of skew-symmetric formulations

In order to obtain energy stable numerical schemes for non-linear problems or for linear problems with variable coefficients, a well-known approach is to rewrite the underlying equations into a split formulation which deviates from the classical divergence form. By applying an SBP scheme to this modified equation, it is possible to construct a simultaneously energy stable and conservative scheme. In particular, this idea has been applied to the inviscid Burgers' equation of which the divergence form is given by

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) = 0, \quad \alpha \leq x \leq \beta, \quad t > 0, \quad (1.22)$$

with non-linear flux function  $f(u) = \frac{1}{2}u^2$ . The so-called *skew-symmetric* inviscid Burgers' equation, which is analytically equivalent to (1.22), is a specific split formulation given by

$$\frac{\partial}{\partial t}u(x,t) + \frac{2}{3}\frac{\partial}{\partial x}f(u(x,t)) + \frac{1}{3}u(x,t)\frac{\partial}{\partial x}u(x,t) = 0, \quad \alpha \leq x \leq \beta, \quad t > 0. \quad (1.23)$$

This reformulation allows for an easy proof of a continuous energy estimate, since multiplication of (1.23) by  $u(x,t)$  and subsequent integration over the computational domain  $[\alpha, \beta]$  yields

$$\begin{aligned} \frac{d}{dt}\left(\frac{1}{2}\|u(x,t)\|_{L^2(\alpha,\beta)}^2\right) &= -\frac{1}{3}\left(\int_\alpha^\beta u(x,t)\frac{\partial}{\partial x}u^2(x,t)dx + \int_\alpha^\beta u^2(x,t)\frac{\partial}{\partial x}u(x,t)dx\right) \\ &= -\frac{1}{3}\int_\alpha^\beta \frac{\partial}{\partial x}u^3(x,t)dx \\ &= \frac{1}{3}(u^3(\alpha,t) - u^3(\beta,t)). \end{aligned} \quad (1.24)$$

In particular, equation (1.24) shows that the energy  $\eta(u) = \frac{1}{2}u^2$  is a conserved quantity of the inviscid Burgers' equation (in fact  $\eta$  is also an entropy function of (1.22)).

In the second step of the derivation (1.24), integration by parts was employed, suggesting that the above energy estimate carries over to the semi-discrete case if an SBP operator is applied. The discretization of (1.23) by an SBP scheme with diagonal norm  $\mathbf{M}$  yields

$$\frac{d\mathbf{u}}{dt} + \frac{1}{3}\mathbf{D}\mathbf{u}^2 + \frac{1}{3}\underline{\mathbf{u}}\mathbf{D}\mathbf{u} = 0, \quad (1.25)$$

where a short notation for pointwise operations is employed, i.e. denoting  $\mathbf{u}^2 = (u_0^2, \dots, u_N^2)^T$  and the diagonally inserted nodal values by  $\underline{\mathbf{u}} = \text{diag}(\mathbf{u})$ . Boundary conditions and corresponding SAT terms are neglected in this example.

The SBP property compensates for the lack of a discrete product rule, since in general,

$$\mathbf{D}\underline{\mathbf{u}}\mathbf{v} \neq \underline{\mathbf{u}}\mathbf{D}\mathbf{v} + \underline{\mathbf{v}}\mathbf{D}\mathbf{u}$$

for arbitrary values of nodal values  $\mathbf{u}, \mathbf{v}$ . Instead we make use of the discrete integration-by-parts rule fulfilled by the SBP scheme and multiply equation (1.25) from the left by  $\mathbf{u}^T\mathbf{M}$  to obtain

$$\frac{1}{2}\frac{d}{dt}\|\mathbf{u}\|_{\mathbf{M}}^2 + \frac{1}{3}\mathbf{u}^T\mathbf{M}\mathbf{D}\mathbf{u}^2 + \frac{1}{3}\mathbf{u}^T\underline{\mathbf{u}}\mathbf{M}\mathbf{D}\mathbf{u} = 0, \quad (1.26)$$

since the diagonal matrices  $\mathbf{M}$  and  $\underline{\mathbf{u}}$  commute. Using the SBP property (1.4) as well as the identity  $\mathbf{u}^T\underline{\mathbf{u}} = (\mathbf{u}^2)^T$ , we have

$$\frac{1}{2}\frac{d}{dt}\|\mathbf{u}\|_{\mathbf{M}}^2 + \frac{1}{3}\mathbf{u}^T(\mathbf{B} - \mathbf{D}^T\mathbf{M})\mathbf{u}^2 + \frac{1}{3}(\mathbf{u}^2)^T\mathbf{M}\mathbf{D}\mathbf{u} = 0, \quad (1.27)$$

Since the scalar quantity  $\mathbf{u}^T\mathbf{D}^T\mathbf{M}\mathbf{u}^2$  equals its transpose  $(\mathbf{u}^T\mathbf{D}^T\mathbf{M}\mathbf{u}^2)^T = (\mathbf{u}^2)^T\mathbf{M}\mathbf{D}\mathbf{u}$ , which cancels out the third term on the right-hand side of (1.27), this further simplifies to

$$\frac{1}{2}\frac{d}{dt}\|\mathbf{u}\|_{\mathbf{M}}^2 = -\frac{1}{3}\mathbf{u}^T\mathbf{B}\mathbf{u}^2 = \frac{1}{3}(u_0^3 - u_N^3), \quad (1.28)$$

The resulting equation (1.28) signifies that a change of discrete energy, particularly its growth, can only occur due to boundary conditions controlling the values at the boundary nodes  $x_0 = \alpha$  and  $x_N = \beta$ . This semi-discrete energy estimate for a diagonal-norm SBP scheme exactly mimics the continuous energy behavior given in (1.24).

A distinct advantage of discretizing the skew-symmetric inviscid Burgers' equation by an SBP scheme is that this approach still yields a conservative numerical method even though the underlying PDE is not written in divergence form. In order to prove this for the present example, we multiply equation (1.25) by  $\mathbf{1}^T \mathbf{M}$  to obtain

$$\begin{aligned} \frac{d}{dt} (\mathbf{1}^T \mathbf{M} \mathbf{u}) &= -\frac{1}{3} (\mathbf{1}^T \mathbf{M} \mathbf{D} \mathbf{u}^2 + \mathbf{u}^T \mathbf{M} \mathbf{D} \mathbf{u}) \\ &\stackrel{(1.4)}{=} -\frac{1}{3} \left( \mathbf{1}^T (\mathbf{B} - \mathbf{D}^T \mathbf{M}) \mathbf{u}^2 + \frac{1}{2} \mathbf{u}^T (\mathbf{B} - \mathbf{D}^T \mathbf{M} + \mathbf{M} \mathbf{D}) \mathbf{u} \right). \end{aligned}$$

Hereby, except for the boundary contributions, all the terms on the right-hand side vanish. More precisely, we have  $\mathbf{1}^T \mathbf{D}^T \mathbf{M} = (\mathbf{D} \mathbf{1})^T \mathbf{M} = \mathbf{0} = (0, \dots, 0)^T$  if the SBP operator  $\mathbf{D}$  is consistent, meaning that in Definition 1.1, the degree condition for  $q = 0$  is fulfilled, and the last two terms are equal by transposition of scalars, i.e.  $\mathbf{u}^T \mathbf{D}^T \mathbf{M} \mathbf{u} = (\mathbf{u}^T \mathbf{D}^T \mathbf{M} \mathbf{u})^T = \mathbf{u}^T \mathbf{M} \mathbf{D} \mathbf{u}$ . Therefore, we obtain

$$\frac{d}{dt} (\mathbf{1}^T \mathbf{M} \mathbf{u}) = -\frac{1}{3} \left( \mathbf{1}^T \mathbf{B} \mathbf{u}^2 + \frac{1}{2} \mathbf{u}^T \mathbf{B} \mathbf{u} \right) = \frac{1}{2} (u_0^2 - u_N^2). \quad (1.29)$$

Since  $\mathbf{1}^T \mathbf{M} \mathbf{u} \approx \int_{\alpha}^{\beta} u(x, t) dx$  accurately approximates the integrated quantity  $u(x, t)$ , equation (1.29) signifies that a change in time of the discrete mass given by  $\mathbf{1}^T \mathbf{M} \mathbf{u}$  only occurs via fluxes through the boundaries of the domain at  $x_0 = \alpha$  and  $x_N = \beta$ . This again mimics the conservation property which is inherent to the inviscid Burgers' equation.

Naturally, an SBP scheme applied to the divergence form of Burgers' equation (1.22) also yields discrete conservation. In fact, any discrete divergence operator having the SBP property yields discrete conservation of the primary conserved quantities if the given conservation law is in divergence form. In case of the Euler equations of gas dynamics or the compressible Navier-Stokes equations as particular examples, these primary quantities are mass, momentum and total energy. The decided advantage of combining SBP schemes with certain split forms, e.g. skew-symmetric formulations, is that even if applied to these specific fluid equations which are not in divergence form, conservation of the primary quantities is achieved. Moreover, the additional conservation of specific secondary quantities such as kinetic energy or entropy for the Euler equations may be obtained when choosing a suitable split formulation.

Based on a specific skew-symmetric formulation of the Euler equations, a kinetic energy preserving DG scheme on Legendre-Gauss nodes is constructed in Section 1.4, profiting from the generalized SBP property of the corresponding first derivative operator.

### Connection to finite volume schemes

While SBP properties have been recognized in certain types of finite volume schemes, see e.g. [143], vice versa, finite difference SBP schemes have been rewritten as finite volume

schemes by Fisher et al. in [56]. More precisely, Fisher et al. rewrite finite difference SBP operators in telescoping sum formulation and the resulting schemes can be recognized as finite volume schemes on suitably defined subcells. This property is advantageous in case of shocks occurring in the exact solution. In fact, for a finite difference scheme in conservative form which is applied to a conservation law in divergence form, the well-known Theorem of Lax and Wendroff may be applied. This theorem guarantees that if the scheme is convergent, then it converges to a weak solution and hence yields correct shock speeds. However, it is not necessary to derive the discrete operator from the divergence form of the continuous equations. On the contrary, carefully designed discrete operators applied to linearly split forms of the conservation law are equivalent to telescoping operators fulfilling the assumptions of the Lax-Wendroff theorem, as proven in [56]. Hereby, the subcell finite volume property plays an important role in this context.

### Generalized SBP schemes

In [47], Del Rey Fernández et al. provided a generalized framework for SBP operators in one space dimension which extends the classical finite difference background. Thereby, a wide range of operators may be considered to be SBP operators, e.g. nodal pseudo-spectral operators on Legendre-Gauss, Legendre-Gauss-Lobatto or Legendre-Gauss-Radau points. The main extensions compared to classical finite difference SBP operators are non-repeating interior point operators, non-uniform nodal distributions within the computational domain and nodal distributions which do not include one or both boundary points. The concept has later been transferred to multidimensional domains including simplex elements in [77]. The basic insight for the construction of generalized SBP operators is that the existence of a quadrature rule is necessary and sufficient for the existence of an SBP operator, whereby its degree is intimately linked to the degree of the underlying quadrature rule. Therefore, it is possible to derive new SBP operators by first constructing suitably accurate quadrature rules and subsequently solving the degree conditions for the derivative operator  $\mathbf{D}$ . The generalization in [47] furthermore allows for the construction of SBP operators in a finite element setting where a refinement consists in increasing the number of elements which carry a fixed nodal distribution in contrast to increasing the number of equidistant nodes as in finite difference schemes. Hence, quite general types of discontinuous Galerkin schemes may also be seen as SBP schemes.

A generalized SBP finite difference operator approximating the first derivative  $\frac{\partial}{\partial x}$  can be defined on arbitrary non-uniform nodal distributions on the domain  $[\alpha, \beta]$  which may or may not include the domain boundaries. Denoting these grid points by  $x_j$ ,  $j = 0, \dots, N$ , analogously to the equidistant case, an approximate solution to the 1D linear advection equation (1.1) is then defined by the corresponding time-dependent solution vector

$$\mathbf{u}(t) = (u_0(t), \dots, u_N(t))^T \approx (u(x_0, t), \dots, u(x_N, t))^T$$

on the specific nodal distribution. Furthermore, the Definition 1.1 of a classical SBP scheme is extended as follows, see [47], Definition 2.

**Definition 1.3.** *A generalized SBP (GSBP) finite difference scheme to solve (1.1) is of the*

form

$$\frac{d\mathbf{u}}{dt} + a\mathbf{D}\mathbf{u} = \sigma\mathbf{M}^{-1}\mathbf{t}_\alpha \mathbf{t}_\alpha^T \mathbf{u}, \quad (1.30)$$

where  $\mathbf{t}_\alpha$  is a projection to the left boundary defined below and the SAT parameter  $\sigma \in \mathbb{R}$  has to be specified to obtain energy stability – analogously to the classical finite difference SBP scheme given in (1.3).

The scheme (1.30) is a generalized SBP scheme with first-derivative SBP operator  $\mathbf{D}$  of degree  $q$  if the subsequent conditions are fulfilled.

1. The matrix  $\mathbf{D}$  is an accurate approximation to  $\frac{\partial}{\partial x}$  with

$$\mathbf{D}\mathbf{x}^k = k\mathbf{x}^{k-1}, \quad 0 \leq k \leq q,$$

where  $\mathbf{x}^k = (x_0^k, \dots, x_N^k)^T$  is the representation of the monomials  $x^k$  on the grid points.

2. The matrix  $\mathbf{M}$  is symmetric and positive definite.
3. Setting  $\mathbf{S} = \mathbf{M}\mathbf{D}$ , integration by parts is mimicked by the property

$$\mathbf{S} + \mathbf{S}^T = \mathbf{M}\mathbf{D} + \mathbf{D}^T\mathbf{M} = \mathbf{B} = \mathbf{B}^T = \mathbf{t}_\beta \mathbf{t}_\beta^T - \mathbf{t}_\alpha \mathbf{t}_\alpha^T, \quad (1.31)$$

where  $\mathbf{B}$  is an interface and boundary operator with the property  $(\mathbf{x}^l)^T \mathbf{B} \mathbf{x}^m = [x^{l+m}]_\alpha^\beta$  for all  $0 \leq l, m \leq r$ , where  $r \geq q$  denotes the degree of the SAT terms used for imposition of boundary and interface conditions. Furthermore,  $\mathbf{B}$  decomposes into the left and right boundary contributions  $\mathbf{t}_\alpha \mathbf{t}_\alpha^T$  and  $\mathbf{t}_\beta \mathbf{t}_\beta^T$ , i.e.

$$\mathbf{B} = \mathbf{t}_\beta \mathbf{t}_\beta^T - \mathbf{t}_\alpha \mathbf{t}_\alpha^T, \quad (1.32)$$

where  $\mathbf{t}_\alpha^T \mathbf{x}^l = \alpha^l$  and  $\mathbf{t}_\beta^T \mathbf{x}^l = \beta^l$  for  $0 \leq l \leq r$ .

The concept of generalized SBP operators has also been extended to the second derivative in order to solve viscous flow equations. The corresponding definition in [49] extends Definition 1.2 in a straightforward manner, where the generalized discrete second-derivative operator  $\mathbf{D}_2$  is defined as in (1.20) but operates on potentially non-equidistant grid points not including the domain boundaries and the boundary operator  $\mathbf{B}$  is taken from Definition 1.3. Analogously to classical SBP operators, second-derivative GSBP operators may be extended to second derivatives with variable coefficients and it is possible to construct order-matched operators where first-derivative and second-derivative operators are of the same order.

### Upwind SBP schemes

Recently, Mattsson [126] has derived a class of finite difference methods called *upwind SBP schemes*. These schemes consist of dual-pair SBP operators with non-central difference stencils which lead to a built-in artificial dissipation when combined with flux splitting. This approach serves to stabilize non-linear problems while retaining the linear stability properties associated with SBP schemes.

**Definition 1.4.** A pair of difference operators denoted by  $\mathbf{D}^+$  and  $\mathbf{D}^-$ , both approximating the first derivative  $\frac{\partial}{\partial x}$ , are called diagonal-norm upwind SBP operators if

1. For a positive definite diagonal matrix  $\mathbf{M}$ , the derivative operators are given by

$$\mathbf{D}^- = \mathbf{M}^{-1} \left( \mathbf{Q}^- + \frac{1}{2} \mathbf{B} \right) \text{ and } \mathbf{D}^+ = \mathbf{M}^{-1} \left( \mathbf{Q}^+ + \frac{1}{2} \mathbf{B} \right), \quad (1.33)$$

with  $\mathbf{B} = \text{diag}(-1, 0, \dots, 0, 1)$ .

2. Integration by parts is mimicked by the property

$$\mathbf{Q}^+ + (\mathbf{Q}^-)^T = \mathbf{0}. \quad (1.34)$$

3. As an additional stability constraint, the symmetric matrix

$$\mathbf{C} := \frac{1}{2} (\mathbf{Q}^+ - \mathbf{Q}^-) = \frac{1}{2} \left( \mathbf{Q}^+ + (\mathbf{Q}^+)^T \right) = -\frac{1}{2} \left( \mathbf{Q}^- + (\mathbf{Q}^-)^T \right) \quad (1.35)$$

is negative semi-definite.

The motivation for using upwind SBP operators is the additional artificial damping introduced by enforcing negative semi-definiteness of the matrix  $\mathbf{C}$  in (1.35). We will illustrate this property of an upwind SBP scheme in comparison to classical SBP schemes in the following example.

Reconsidering the discretization of the linear advection equation (1.1) via an SBP scheme, we replace the SBP operator  $\mathbf{D}$  in (1.3) by the upwind SBP operator  $\mathbf{D}^-$  (for  $a < 0$ , the operator  $\mathbf{D}^+$  would be applied). Keeping the SAT term, we obtain the discretization

$$\frac{d\mathbf{u}}{dt} + a\mathbf{D}^- \mathbf{u} = \sigma \mathbf{M}^{-1} \mathbf{e}_0 u_0, \quad \mathbf{e}_0 = (1, 0, \dots, 0)^T. \quad (1.36)$$

Multiplying (1.36) from the left by  $\mathbf{u}^T \mathbf{M}$  and adding the transpose now yields

$$2\mathbf{u}^T \mathbf{M} \frac{d\mathbf{u}}{dt} + a\mathbf{u}^T \left( \mathbf{M} \mathbf{D}^- + (\mathbf{D}^-)^T \mathbf{M} \right) \mathbf{u} = 2\sigma u_0^2. \quad (1.37)$$

Using the upwind SBP properties (1.34) and (1.35), we have

$$\mathbf{M} \mathbf{D}^- + (\mathbf{D}^-)^T \mathbf{M} = \mathbf{Q}^- + (\mathbf{Q}^-)^T + \mathbf{B} = -2\mathbf{C} + \mathbf{B},,$$

and therefore

$$\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 + a(u_N^2 - u_0^2) = 2\sigma u_0^2 + 2\mathbf{u}^T \mathbf{C} \mathbf{u}.$$

For  $\sigma \leq -\frac{a}{2}$ , the time evolution of the discrete energy may now be estimated by

$$\frac{d}{dt} \|\mathbf{u}\|_{\mathbf{M}}^2 + au_N^2 \leq 2\mathbf{u}^T \mathbf{C} \mathbf{u}, \quad (1.38)$$

where the last term containing the negative semi-definite matrix  $\mathbf{C}$  introduces additional artificial damping.

Upwind SBP operators also enable the construction of diagonal-norm second-derivative operators of the form

$$\begin{aligned}\mathbf{D}_2^- &= \mathbf{D}^+ \mathbf{D}^- = \mathbf{M}^{-1}(-(\mathbf{D}^-)^T \mathbf{M} \mathbf{D}^- + \mathbf{B} \mathbf{D}^-), \\ \mathbf{D}_2^+ &= \mathbf{D}^- \mathbf{D}^+ = \mathbf{M}^{-1}(-(\mathbf{D}^+)^T \mathbf{M} \mathbf{D}^+ + \mathbf{B} \mathbf{D}^+),\end{aligned}\quad (1.39)$$

similar to Definition 1.2. We will derive second-derivative operators of this form for specific DG diffusion discretizations in Section 2.1.

## 1.2 The DG scheme in SBP framework

Discontinuous Galerkin schemes, first mainly advanced by Cockburn and Shu in [43] and references therein, are based on a variational formulation similar to classical Galerkin finite element schemes with coincident ansatz and test spaces. In contrast to the continuity assumptions of classical FE methods, DG schemes allow for piece-wise smooth approximate solutions with potential discontinuities across element interfaces. Due to its flexibility and generality, the DG scheme is a popular numerical method in a variety of applications ranging from compressible fluid flow and aeroacoustics [180, 50, 35] to electromagnetics [103, 45], meteorology [71, 167] and geophysics [55]. The main advantages of the DG approach are its local conservation property, an arbitrarily high order of accuracy and superconvergence capabilities. Dispensing with global continuity of the approximate solution, the DG approach results in relatively compact stencils which greatly facilitates both hp-adaptivity of the method and its implementation in parallel hardware environment.

In recent investigations, connections between modern nodal DG schemes and classical SBP finite difference methods have been established with the objective to take advantage of this connection to discretize conservation laws in skew-symmetric form and to obtain certain energy estimates. This section deals with the SBP properties of nodal DG schemes in one space dimension in Sections 1.2.1 and 1.2.3, on tensor-product grids in Section 1.2.4 and on unstructured triangular grids in Section 1.2.5 providing the link to the definition of generalized and upwind SBP operators introduced in Section 1.1. Furthermore, in Section 1.2.2, we introduce the closely related subcell finite volume property of DG schemes on interior nodal distributions, rewriting the DG scheme as a finite volume formulation on a division of each DG cell into a certain number of subcells.

### 1.2.1 The one-dimensional case

In order to verify the SBP property of classical 1D-DG schemes, we consider a scalar hyperbolic conservation law in one space dimension given by

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) = 0, \quad t > 0, \quad x \in \Omega = [a, b] \subset \mathbb{R}. \quad (1.40)$$

After a subdivision of the spatial domain  $\Omega$  into sub-intervals,  $\Omega = \bigcup_{i=1}^K I_i = \bigcup_{i=1}^K [x_i, x_{i+1}]$ , the DG scheme constructs an approximation  $u_h$  of  $u$  which is piece-wise continuous which means that on each sub-interval  $I_i$  we have

$$u_h(x, t)|_{I_i} = u_h^i(x, t) = \sum_{j=1}^{N+1} u_j^i(t) \Phi_j^i(x) \quad (1.41)$$

using basis functions  $\Phi_j^i$ , usually given by polynomial functions  $\Phi_j^i \in \mathcal{P}^N([x_i, x_{i+1}])$ . Hereby, we will generally consider the simplified case that the polynomial degree  $N$  is fixed throughout the computational domain, although the flexibility of the DG scheme allows to chose different polynomial degrees on different DG cells.

The DG scheme in weak form on a certain sub-interval  $I_i$  is obtained as usual by first multiplying the conservation law (1.40) with global test functions  $\tilde{\Phi}_j^i$  constructed from the local ones by

$$\tilde{\Phi}_j^i(x) = \begin{cases} \Phi_j^i(x) & \text{if } x \in I_i, \\ 0 & \text{otherwise.} \end{cases}$$

Second, partial integration of the term containing the flux  $f$  is carried out and a numerical flux function is introduced to ensure coupling between the sub-intervals. This procedure yields the DG scheme in weak form

$$\frac{d}{dt} \int_{I_i} u_h \Phi_k^i dx + f_i^* \Phi_k^i(x_i) - f_{i+1}^* \Phi_k^i(x_{i+1}) - \int_{I_i} f(u_h) \frac{d\Phi_k^i}{dx} dx = 0, \quad (1.42)$$

with  $f_i^* = f^*(u_h^{i-1}(x_i), u_h^i(x_i))$  denoting the values of a consistent numerical flux function  $f^*$ . The numerical flux hereby requires the left and right-hand side limits as input arguments which are given by the values  $u_h^{i-1}(x_i), u_h^i(x_i)$  as defined in (1.41) while neglecting the dependence on the time variable  $t$ . The last integral term on the left hand side of equation (1.42) is generally not solved analytically but by numerical quadrature rules. Thus, for numerical integration we consider a set of quadrature nodes

$$\xi_\nu \in [-1, 1], \quad \nu = 1, \dots, N+1,$$

transformed to the specific sub-interval under consideration. The corresponding weights will be denoted by

$$\omega_\nu, \quad \nu = 1, \dots, N+1.$$

Hereby, we restrict the presentation to nodal sets containing exactly  $\dim \mathcal{P}^N = N+1$  points. In this case, we may as well use these nodes to construct an interpolation  $f_h \in \mathcal{P}^N$  of the nonlinear function  $f(u_h)$ . Exact integration of  $\int_{I_i} f_h \frac{d\Phi_k^i}{dx} dx$  is then equivalent to numerical integration of the last integral in (1.42) for an arbitrary function  $f$  if and only if the quadrature rule is exact for polynomials of degree  $2N-1$ .

To construct the DG scheme and highlight its SBP property, the test and basis functions for the expansion of  $u_h$  and within the variational formulation (1.42) are chosen as the corresponding nodal Lagrange polynomials  $\Phi_k^i = L_k^i$  with  $L_k^i(\Lambda_i(\xi_\nu)) = \delta_{\nu k}$ , where  $\Lambda_i$  denotes the transformation of the reference cell  $[-1, 1]$  to the specific sub-interval  $I_i$ , i.e.

$$\Lambda_i(\xi) = \xi \frac{x_{i+1} - x_i}{2} + \frac{x_i + x_{i+1}}{2}. \quad (1.43)$$

Hence, considering the Lagrange polynomials  $L_k : [-1, 1] \rightarrow \mathbb{R}$  corresponding to the quadrature nodes on the reference interval  $[-1, 1]$ , the basis functions with respect to a specific cell are given by  $L_k^i = L_k \circ \Lambda_i^{-1}$ . For a specific cell  $I_i$ , the weak form of the DG scheme using quadrature rules and Lagrange basis functions is then given by

$$\frac{d}{dt} \int_{I_i} u_h L_k^i dx + f_i^* L_k^i(x_i) - f_{i+1}^* L_k^i(x_{i+1}) - \int_{I_i} f_h \frac{dL_k^i}{dx} dx = 0, \quad (1.44)$$

where  $f_h$  is defined by the expansion

$$f_h(x, t)|_{I_i} = \sum_{j=1}^{N+1} f(u_j^i) L_j^i(x), \quad (1.45)$$

with point-wise values  $u_j^i = u_h(\Lambda_i(\xi_j), t)$ . Transforming to the reference cell  $[-1, 1]$ , we obtain the corresponding weak form

$$\begin{aligned} \frac{\Delta x_i}{2} \frac{d}{dt} \int_{-1}^1 u_h(\Lambda_i(\xi), t) L_k(\xi) d\xi + f_i^* L_k(-1) - f_{i+1}^* L_k(1) \\ - \int_{-1}^1 f_h(\Lambda_i(\xi), t) L'_k(\xi) d\xi = 0, \end{aligned} \quad (1.46)$$

where  $\Delta x_i = x_{i+1} - x_i$ .

So far, the integrals appearing in (1.46) are solved exactly since the required numerical integration of the nonlinear term  $\int_{I_i} f(u_h) \frac{d\Phi_k^i}{dx} dx$  in (1.42) has been reinterpreted via interpolation and the subsequent use of a numerical quadrature which exactly integrates polynomials of degree  $2N - 1$ .

Equation (1.46) rewrites in a simpler form using matrix-vector notation. For this purpose, we define the matrices  $\mathbf{M}$  and  $\mathbf{S}$  by their entries

$$M_{jk} = \int_{-1}^1 L_j L_k d\xi = M_{kj}, \quad (1.47)$$

$$S_{jk} = \int_{-1}^1 L_j L'_k d\xi, \quad (1.48)$$

as well as the solution vector  $\mathbf{u}^i$  and vector of flux values  $\mathbf{f}^i$  given by

$$\begin{aligned} \mathbf{u}^i &= (u_1^i, \dots, u_{N+1}^i)^T \text{ with } u_j^i = u_h(\Lambda_i(\xi_j), t), \\ \mathbf{f}^i &= (f_1^i, \dots, f_{N+1}^i)^T \text{ with } f_j^i = f(u_j^i). \end{aligned}$$

Furthermore, we use the abbreviations  $f^{*,i}(1) = f_{i+1}^*$  and  $f^{*,i}(-1) = f_i^*$  and collect the basis functions in the vector valued function

$$\mathbf{L}(\xi) = (L_1(\xi), \dots, L_{N+1}(\xi))^T. \quad (1.49)$$

Using the expansions of  $u_h$  and  $f_h$  into the Lagrange basis functions as in (1.41) and (1.45), the above variational form (1.46) is then equivalent to the matrix-vector form

$$\frac{\Delta x_i}{2} \mathbf{M} \frac{d\mathbf{u}^i}{dt} - \mathbf{S}^T \mathbf{f}^i = -[f^{*,i} \mathbf{L}]_{-1}^1.$$

The DG scheme in strong form is obtained by a second partial integration of (1.44) resulting in the variational formulation

$$\int_{I_i} \frac{\partial u_p}{\partial t} L_k^i dx + \int_{I_i} \frac{\partial f_h}{\partial x} L_k^i dx = [f_i^* - f_h^i(x_i)] L_k^i(x_i) - [f_{i+1}^* - f_h^i(x_{i+1})] L_k^i(x_{i+1}). \quad (1.50)$$

The above equation is the equivalent to the matrix-vector formulation

$$\frac{\Delta x_i}{2} \mathbf{M} \frac{d\mathbf{u}^i}{dt} + \mathbf{S} \mathbf{f}^i = [(f_h^i - f^{*,i}) \mathbf{L}]_{-1}^1,$$

where  $f_h^i(\xi, t) = \sum_{j=1}^{N+1} f(u_j^i) L_j(\xi)$  is the transformation of  $f_h(x, t)|_{I_i}$  to the reference cell, i.e.  $f_h^i(\xi, t) = f_h(\Lambda_i(\xi), t)$ . For pairwise distinct nodes, the Lagrange polynomials represent a set of linearly independent functions. Hence, the matrix  $\mathbf{M}$ , given by the entries (1.47) is symmetric and positive definite and thus invertible. The relation to SBP schemes can now be observed by multiplying with the inverse of  $\mathbf{M}$  to obtain

$$\frac{\Delta x_i}{2} \frac{d\mathbf{u}^i}{dt} + \mathbf{D} \mathbf{f}^i = \mathbf{M}^{-1} [(f_h^i - f^{*,i}) \mathbf{L}]_{-1}^1. \quad (1.51)$$

where the entries of  $\mathbf{D} = \mathbf{M}^{-1} \mathbf{S}$  are given by

$$D_{jk} = L'_k(\xi_j), \quad (1.52)$$

due to the interpolation property of the Lagrange polynomials. In fact, defining the matrix  $\mathbf{D}$  by the entries (1.52) leads to

$$(\mathbf{M} \mathbf{D})_{ij} = \sum_k M_{ik} D_{kj} = \int_{-1}^1 L_i(\xi) \sum_k L_k(\xi) L'_j(\xi_k) d\xi = \int_{-1}^1 L_i(\xi) L'_j(\xi) d\xi = S_{ij}.$$

The above form (1.51) of the DG scheme is also obtained when the same numerical quadrature rule is also used for the computation of the integrals in (1.47) defining the entries of  $\mathbf{M}$ . This results in a slightly modified mass matrix  $\mathbf{M}$  in (1.51) which is diagonal and often called a *lumped mass matrix*. More precisely, we then have

$$\int_{-1}^1 L_j L_k d\xi \approx M_{jk} = \sum_l \omega_l L_j(\xi_l) L_k(\xi_l) = \omega_j \delta_{jk}.$$

The entries of  $\mathbf{D}$  remain unmodified since the entries of  $\mathbf{S}$  are exactly integrated in (1.48), and we have

$$(\mathbf{M} \mathbf{D})_{ij} = \omega_i L'_j(\xi_i) = \sum_k \omega_k L_i(\xi_k) L'_j(\xi_k) = \int_{-1}^1 L_i(\xi) L'_j(\xi) d\xi = S_{ij}.$$

When we consider the DG scheme in cell-wise fashion, it may become convenient to drop the cell index  $i$  in equation (1.51) by writing

$$\frac{\Delta x}{2} \frac{d\mathbf{u}}{dt} + \mathbf{D}\mathbf{f} = \mathbf{M}^{-1}[(f_h - f^*)\mathbf{L}]_{-1}^1. \quad (1.53)$$

However, when referring to contributions of interface fluxes, the corresponding interface indices  $i$  and  $i + 1$  referring to the grid nodes  $x_i$  and  $x_{i+1}$  will still be used as in (1.46) and (1.50).

For the DG scheme (1.53) written in terms of the spatial variable  $\xi$  on the reference cell  $[-1, 1]$  an SBP property of the generalized form given in Definition 1.3 is proven in the following Theorem.

**Theorem 1.5.** *The DG scheme (1.53) is an SBP scheme with the SBP operator  $\mathbf{D} = \mathbf{M}^{-1}\mathbf{S}$  given in (1.52). Hereby,  $\mathbf{D}$  approximates  $\frac{\partial}{\partial\xi}$  to degree  $q = N$  and the degree of  $\mathbf{B}$  is  $r = q = N$  as well. Furthermore, given a function  $g(\xi)$  with point-wise values  $\mathbf{g}$ , the interface and boundary operator  $\mathbf{B}$  acts on  $\mathbf{g}$  as*

$$\mathbf{B}\mathbf{g} = [g_h\mathbf{L}]_{-1}^1,$$

where  $g_h = \sum_{j=1}^{N+1} g_j L_j(\xi)$  denotes the polynomial interpolation of the point-wise values  $\mathbf{g}$ .

Therefore, the operator  $\mathbf{B}$  is given by  $\mathbf{B} = [\mathbf{L}\mathbf{L}^T]_{-1}^1$ .

*Proof.* The first task is to prove that  $\mathbf{D}\boldsymbol{\xi}^j = j\boldsymbol{\xi}^{j-1}$  holds for all  $0 \leq j \leq N$ . On this, we consider a polynomial function  $\xi^j$ ,  $j \leq N$ , expanded in Lagrange polynomials in the form  $\xi^j = \sum_{k=1}^{N+1} \xi_k^j L_k(\xi)$ . Since the derivative of  $\xi^j$  is given by the function  $j\xi^{j-1}$ , we have  $j\xi^{j-1} = \sum_k \xi_k^j L'_k(\xi)$ , or else

$$j\boldsymbol{\xi}^{j-1} = \sum_{k=1}^{N+1} \xi_k^j L'_k(\boldsymbol{\xi}), \quad (1.54)$$

in vector notation, collecting the point-wise values of  $\xi^{j-1}$ . Now, an application of the matrix  $\mathbf{D}$  to a vector results in a suitable linear combination of the columns of  $\mathbf{D}$ , i.e.

$$\mathbf{D}\boldsymbol{\xi}^j = \sum_{k=1}^{N+1} \xi_k^j L'_k(\boldsymbol{\xi}) = j\boldsymbol{\xi}^{j-1},$$

where the last equality is due to (1.54). Secondly, the matrix  $\mathbf{M}$  obviously is symmetric and positive definite by construction, as stated before. Lastly, we consider the boundary operator  $\mathbf{B}$ . The entries of  $\mathbf{M}\mathbf{D} + \mathbf{D}^T\mathbf{M}$  are given by

$$(\mathbf{M}\mathbf{D} + \mathbf{D}^T\mathbf{M})_{jk} = S_{jk} + S_{kj} = \int_{-1}^1 (L_j L'_k + L'_j L_k) d\xi = [L_j L_k]_{-1}^1 = B_{jk}.$$

Therefore, we obtain the generalized SBP property given in (1.31) as well as the assertion

$$(\boldsymbol{\xi}^l)^T \mathbf{B} \boldsymbol{\xi}^m = \sum_{j=1}^{N+1} \sum_{k=1}^{N+1} \xi_j^l B_{jk} \xi_k^m = \left[ \sum_{j=1}^{N+1} \xi_j^l L_j(\boldsymbol{\xi}) \cdot \sum_{k=1}^{N+1} \xi_k^m L_k(\boldsymbol{\xi}) \right]_{-1}^1 = [\boldsymbol{\xi}^l \boldsymbol{\xi}^m]_{-1}^1,$$

for all  $0 \leq l, m \leq N$ . Furthermore, (1.32) is fulfilled since  $\mathbf{B}$  decomposes into the left and right boundary contributions  $\mathbf{t}_\alpha \mathbf{t}_\alpha^T = \mathbf{L}(-1)\mathbf{L}^T(-1)$  and  $\mathbf{t}_\beta \mathbf{t}_\beta^T = \mathbf{L}(1)\mathbf{L}^T(1)$ . Finally, denoting the columns of  $\mathbf{B}$  by  $\mathbf{b}_k$ , we have the action of  $\mathbf{B}$  represented by

$$\mathbf{B} \mathbf{g} = \sum_{k=1}^{N+1} \mathbf{b}_k g_k = \left[ \mathbf{L}(\xi) \sum_{k=1}^{N+1} g_k L_k(\xi) \right]_{-1}^1 = [g_h \mathbf{L}]_{-1}^1,$$

as stated.  $\square$

If the numerical quadrature rule integrating the nonlinear term in (1.42) exactly integrates polynomials of degree  $2N$ , no difference is obtained for the two variants of defining the mass matrix  $\mathbf{M}$ , either if it is obtained via exact integration or if (1.47) is replaced by the chosen quadrature rule. As discussed before,  $\mathbf{M}$  is then diagonal since we have  $M_{jk} = \delta_{jk}\omega_j$ . In this case, the discrete derivative operator  $\mathbf{D}$  is called a diagonal-norm SBP operator.

However, we note that so far only a quadrature rule of degree  $2N - 1$  has been assumed. Thus  $\mathbf{M}$  is either obtained by exact integration, potentially yielding a non-diagonal mass matrix, or via the chosen quadrature rule, resulting in mass lumping.

Non-diagonal matrices  $\mathbf{M}$  yield so-called non-diagonal norm SBP operators while mass lumping again results in a diagonal mass matrix with  $M_{jk} = \delta_{jk}\omega_j$  and a diagonal norm SBP operator. For the case of Legendre-Gauss-Lobatto nodes, Gassner has shown this property of the DG scheme with mass lumping in [61].

Choosing the classical Legendre-Gauss nodes which do not contain the interval end points  $\xi = -1$  and  $\xi = 1$  yields exact integration of polynomials up to degree  $2N + 1$ . Due to the improved accuracy of the resulting DG scheme, these points might be preferred to the Legendre-Gauss-Lobatto variant with mass lumping. Higher efficiency of Legendre-Gauss nodes especially for a non-linear example based on the two-dimensional Euler equations is numerically demonstrated in [101]. However, as also stated in [101], in addition to the lower cost based on the fact that boundary interpolation is not required, the DG scheme on Legendre-Gauss-Lobatto nodes also allows larger time steps in case of explicit time integration. Time steps may be taken roughly twice as large in comparison to Legendre-Gauss nodes as shown in [65]. Further subtleties arise as Legendre-Gauss integration may increase robustness for non-linear problems and underresolved simulations, see e.g. [65, 12]. Hence, the question of efficiency will depend on the specific application including accuracy requirements. In addition, the situation of reduced exactness for Legendre-Gauss-Lobatto nodes may be different if the full mass matrix  $\mathbf{M}$  defined by exact integration in (1.47) is used. In this case, a technique in [192] lowers the cost of applying the inverse of  $\mathbf{M}$  to an arbitrary vector. Matrix-vector multiplication then results in an  $\mathcal{O}(N)$  operation, i.e. it becomes as expensive as mass lumping. However, enforcing a balance of certain secondary quantities such as kinetic energy as discussed in Section 1.4.1 might again necessitate a diagonal mass matrix.

Differences with respect to the nodal sets used within the DG scheme also arise in terms of the boundary operator  $\mathbf{B}$ . If both endpoints of the reference cell are included in the DG nodal set, we have a diagonal interface and boundary operator  $\mathbf{B} = \text{diag}\{-1, 0, \dots, 0, 1\}$ . In general,  $\mathbf{B}$  is non-diagonal. For example, in case of Legendre-Gauss (LG) nodes, the specific forms of the interface and boundary operator  $\mathbf{B}$  for  $N = 1, 2$  are given by

$$\mathbf{B}_{LG,N=1} = \text{diag}\{-\sqrt{3}, \sqrt{3}\}, \quad \mathbf{B}_{LG,N=2} = \begin{pmatrix} -\frac{1}{\xi^3} & \frac{1-\xi^2}{\xi^3} & 0 \\ \frac{1-\xi^2}{\xi^3} & 0 & \frac{\xi^2-1}{\xi^3} \\ 0 & \frac{\xi^2-1}{\xi^3} & \frac{1}{\xi^3} \end{pmatrix}, \text{ with } \xi = \sqrt{\frac{3}{5}}.$$

### 1.2.2 The subcell finite volume property

In particular for the DG scheme on Legendre-Gauss-Lobatto nodes with mass lumping, reference is often made to its so-called subcell finite volume property, see e.g. [69].

However, such a property holds for any nodal DG scheme (1.53) with diagonal mass matrix  $\mathbf{M}$ . More precisely, we may rewrite such a DG scheme in the following form

$$\frac{\Delta x}{2} \frac{d}{dt} u_j + \omega_j^{-1} (\bar{f}_{j+1} - \bar{f}_j) = 0, \quad (1.55)$$

where the flux values  $\bar{f}_j$ ,  $j = 1, \dots, N+2$ , need to be suitably defined depending on the DG nodal set.

Equation (1.55) resembles a finite volume scheme on the subcells given by the subcell boundaries  $\bar{x}_j = x_i + \frac{\Delta x}{2} \sum_{\nu=1}^{j-1} \omega_{\nu} \in [x_i, x_{i+1}]$ , where  $[x_i, x_{i+1}]$  denotes the  $i$ th DG cell. An example of such a division in subcells is depicted in Figure 1.1.

In the general case of a nodal DG scheme with diagonal mass matrix, we have the following result which includes the DG scheme on Legendre-Gauss nodes as a special case.

**Lemma 1.6.** *For a general nodal DG scheme of the form (1.53) with diagonal mass matrix  $\mathbf{M}$ , it is possible to rewrite the scheme in the subcell finite volume formulation (1.55) with flux values given by*

$$\bar{f}_1 = f_i^*, \quad (1.56)$$

$$\bar{f}_{j+1} = \bar{f}_j + \omega_j \sum_{\nu=1}^{N+1} L'_{\nu}(\xi_j) f_{\nu} + [(f^* - f_h) L_j]_{-1}^1, \quad j = 1, \dots, N, \quad (1.57)$$

$$\bar{f}_{N+2} = f_{i+1}^*. \quad (1.58)$$

*Proof.* Using the definition of the operators  $\mathbf{D}$  and  $\mathbf{M}$  given in Section 1.2.1, we rewrite the DG scheme (1.53) in terms of each nodal value in the form

$$\frac{\Delta x}{2} \frac{d}{dt} u_j + \sum_{\nu=1}^{N+1} L'_{\nu}(\xi_j) f_{\nu} = \omega_j^{-1} [(f_h - f^*) L_j]_{-1}^1. \quad (1.59)$$

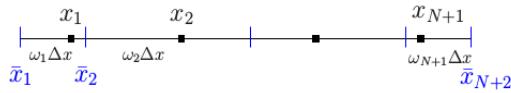


Figure 1.1: Finite volume subcells  $[\bar{x}_j, \bar{x}_{j+1}]$  of length  $\omega_j \Delta x$  and DG nodes  $x_{\nu}$ .

For the first nodal value, by setting

$$\bar{f}_1 = f_i^*, \quad \bar{f}_2 = \omega_1 \sum_{\nu=1}^{N+1} L'_\nu(\xi_1) f_\nu + [(f^* - f_h) L_1]_{-1}^1 + f_i^*,$$

we directly obtain

$$\frac{\Delta x}{2} \frac{d}{dt} u_1 + \omega_1^{-1} (\bar{f}_2 - \bar{f}_1) = \frac{\Delta x}{2} \frac{d}{dt} u_1 + \omega_1^{-1} \left( \omega_1 \sum_{\nu=1}^{N+1} L'_\nu(\xi_1) f_\nu + [(f^* - f_h) L_1]_{-1}^1 \right) = 0$$

from the nodal formulation (1.59). In the same fashion, we have

$$\frac{\Delta x}{2} \frac{d}{dt} u_j + \omega_j^{-1} (\bar{f}_{j+1} - \bar{f}_j) = 0, \quad j = 2, \dots, N+1,$$

if the subsequent subcell finite volume flux values are defined by

$$\bar{f}_{j+1} = \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + [(f^* - f_h) L_j]_{-1}^1 + \bar{f}_j, \quad j = 2, \dots, N+1.$$

It remains to prove that by this construction, the equality  $\bar{f}_{N+2} = f_{i+1}^*$  holds for the last flux value. In fact, inductively, we have

$$\begin{aligned} \bar{f}_{N+2} &= \omega_{N+1} \sum_{\nu=1}^{N+1} L'_\nu(\xi_{N+1}) f_\nu + [(f^* - f_h) L_{N+1}]_{-1}^1 + \bar{f}_{N+1} \\ &= \sum_{j=1}^{N+1} \left( \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + [(f^* - f_h) L_j]_{-1}^1 \right) + f_i^* \\ &= \sum_{\nu=1}^{N+1} f_\nu \int_{-1}^1 L'_\nu(\xi) d\xi + \left[ (f^* - f_h) \sum_{j=1}^{N+1} L_j \right]_{-1}^1 + f_i^* \\ &= \sum_{\nu=1}^{N+1} f_\nu [L_\nu]_{-1}^1 + [f^* - f_h]_{-1}^1 + f_i^* \\ &= [f_h]_{-1}^1 + [f^* - f_h]_{-1}^1 + f_i^* = f_{i+1}^*, \end{aligned}$$

thus, the assertion is proven.  $\square$

According to Lemma 1.6, in general, the flux values  $\bar{f}_j$ ,  $j = 1, \dots, N+2$ , may depend on the values  $f(u_\nu)$ ,  $\nu = 1, \dots, N+1$ , of the flux function evaluated on the DG nodal set and on the values  $f_i^*$ ,  $f_{i+1}^*$  of the numerical flux function at the interfaces. In case of the DG scheme on Legendre-Gauss-Lobatto nodes, the situation is simplified since the interior flux values  $\bar{f}_j$ ,  $j = 2, \dots, N+1$ , do not depend on the values of the numerical flux function while the left and right cell boundary flux values correspond precisely to these values  $f_i^*$  and  $f_{i+1}^*$ . In fact, we have the following Corollary to Lemma 1.6 regarding the DG scheme on Legendre-Gauss-Lobatto nodes.

**Corollary 1.7.** *If the DG scheme (1.53) is defined on Legendre-Gauss-Lobatto nodes and uses Legendre-Gauss-Lobatto quadrature to define the mass matrix  $\mathbf{M}$  via mass lumping, the flux values in (1.55) are given by*

$$\begin{aligned}\bar{f}_1 &= f_i^*, \\ \bar{f}_2 &= f_1 + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_1) f_\nu, \\ \bar{f}_{j+1} &= \bar{f}_j + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu, \quad j = 2, \dots, N, \\ \bar{f}_{N+2} &= f_{i+1}^*,\end{aligned}$$

where  $f_\nu = f(u_\nu)$ ,  $\nu = 1, \dots, N+1$ .

*Proof.* First we note that for the boundary flux values  $\bar{f}_1$  and  $\bar{f}_{N+2}$ , the expressions are the same as in (1.56) and (1.58), respectively, hence there is nothing to prove. Considering the value of  $\bar{f}_2$ , equation (1.57) given in Lemma 1.6 may be rewritten in the following way.

$$\begin{aligned}\bar{f}_2 &= \bar{f}_1 + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + [(f^* - f_h)L_1]_{-1}^1 \\ &= \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + (f_{i+1}^* - f_h(1))L_1(1) + f_h(-1)L_1(-1).\end{aligned}$$

In the case of a nodal DG scheme on Legendre-Gauss-Lobatto nodes, this expression simplifies since we have  $L_1(-1) = 1$ ,  $L_1(1) = 0$  and  $f_h(-1) = f_1$  for this set of nodes. Hence, we obtain

$$\bar{f}_2 = \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + f_1$$

as stated in Corollary 1.7.

Concerning the values of  $\bar{f}_{j+1}$  for  $j = 2, \dots, N$ , the expression (1.57) simplifies due to the fact that the choice of Legendre-Gauss-Lobatto nodes yields vanishing boundary values of the corresponding interior Lagrange polynomials, i.e.  $L_j(\pm 1) = 0$ ,  $j = 2, \dots, N$ . This leads to vanishing boundary terms in (1.57), i.e.

$$\bar{f}_{j+1} = \bar{f}_j + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + 0$$

which proves the assertion of Corollary 1.7 for  $j = 2, \dots, N$ .  $\square$

The following statement indicates how the subcell finite volume flux values  $\bar{f}_j$  are related to the DG discretized flux function  $f_h$  evaluated at the boundaries  $\bar{x}_j$  of the finite volume subcells.

**Lemma 1.8.** *If for a general nodal DG scheme of the form (1.53) with diagonal mass matrix  $\mathbf{M}$ , the following additional assumptions concerning the values of the numerical flux function,*

$$\begin{aligned} f_i^* &= f_h(x_i) + \mathcal{O}((\Delta x)^2), \\ f_{i+1}^* &= f_h(x_{i+1}) + \mathcal{O}((\Delta x)^2), \end{aligned} \quad (1.60)$$

*are fulfilled, then we have estimates*

$$\bar{f}_j = f_h(\bar{x}_j) + \mathcal{O}(\Delta x^2), \quad j = 2, \dots, N+1. \quad (1.61)$$

*for the subcell finite volume flux values in the interior of the DG cell.*

*Furthermore, in case of the DG scheme with mass lumping on Legendre-Gauss-Lobatto nodes, the estimate (1.61) holds without the additional assumptions concerning the values of the numerical flux function.*

**Remark 1.9.** *The above assumptions (1.60) on the accuracy of the values of the numerical flux function are reasonable for a high order DG scheme due to the interpolation accuracy.*

*In fact, using DG nodes to carry out a piecewise interpolation of a smooth function  $u$  on the domain  $[x_{i-1}, x_{i+2}]$  composed of three adjacent cells provides the estimates  $u_h^l(x_i) = u(x_i) + \mathcal{O}((\Delta x)^{N+1})$  with  $l = i-1, i$  and  $u_h^l(x_{i+1}) = u(x_{i+1}) + \mathcal{O}((\Delta x)^{N+1})$  with  $l = i, i+1$ . Furthermore, assuming Lipschitz continuity of the numerical flux function yields both  $f_i^* = f(u(x_i)) + \mathcal{O}((\Delta x)^{N+1})$  and  $f_{i+1}^* = f(u(x_{i+1})) + \mathcal{O}((\Delta x)^{N+1})$ . Using the interpolation property of the polynomial function  $f_h$ , we obtain  $f_h(x_l) = f(u(x_l)) + \mathcal{O}((\Delta x)^{N+1})$ ,  $l = i, i+1$ , which can be reduced to the estimates (1.60). Although the DG solution itself will naturally be less accurate than an interpolation function on the DG nodes, the estimates (1.60) for smooth functions are rather pessimistic for DG schemes with  $N \geq 2$ . Therefore, the assumptions (1.60) are still reasonable considering the DG solution.*

*Proof of Lemma 1.8.* In order to obtain the estimate (1.61), Taylor expansion is used. We have

$$f_h(\bar{x}_{j+1}) = f_h(\bar{x}_j) + (\bar{x}_{j+1} - \bar{x}_j) \frac{df_h}{dx}(x(\xi_j)) + \mathcal{O}(\Delta x^2),$$

with  $\tilde{x} \in [\bar{x}_j, \bar{x}_{j+1}]$ . This yields

$$\begin{aligned} f_h(\bar{x}_{j+1}) &= f_h(\bar{x}_j) + \frac{\Delta x}{2} \omega_j \frac{df_h}{dx}(x(\xi_j)) + \mathcal{O}(\Delta x^2) = f_h(\bar{x}_j) + \omega_j \frac{df_h}{d\xi}(\xi_j) + \mathcal{O}(\Delta x^2) \\ &= f_h(\bar{x}_j) + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + \mathcal{O}(\Delta x^2). \end{aligned} \quad (1.62)$$

First, starting with the assumptions concerning the values of the numerical flux function, we have

$$f_h(\bar{x}_1) = f_i^* + \mathcal{O}(\Delta x^2) = \bar{f}_1 + \mathcal{O}(\Delta x^2)$$

for the leftmost flux value.

The flux values interior to the DG cell may then be dealt with by induction, since the inductive definition of the flux value  $\bar{f}_{j+1}$  in (1.57) and the assumption  $\bar{f}_j = f_h(\bar{x}_j) + \mathcal{O}(\Delta x^2)$  inserted into (1.62) yield

$$\begin{aligned} f_h(\bar{x}_{j+1}) &= \bar{f}_j + \mathcal{O}(\Delta x^2) + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + \mathcal{O}(\Delta x^2) = \bar{f}_{j+1} - [(f^* - f_h)L_j]_{-1}^1 + \mathcal{O}(\Delta x^2) \\ &= \bar{f}_{j+1} + \mathcal{O}(\Delta x^2), \quad j = 1, \dots, N. \end{aligned} \tag{1.63}$$

Hereby, the last equality is again obtained by using the assumptions concerning the values of the numerical flux function.

For the DG scheme with mass lumping on Legendre-Gauss-Lobatto nodes, the situation simplifies and we do not need the additional assumptions. In fact, for this set of nodes we have  $f_h(\bar{x}_1) = f_h(x_1) = f_1$  and we may insert the flux value  $\bar{f}_2$  given in Corollary 1.7 into (1.62) in order to obtain

$$f_h(\bar{x}_2) = f_1 + \omega_1 \sum_{\nu=1}^{N+1} L'_\nu(\xi_1) f_\nu + \mathcal{O}(\Delta x^2) = \bar{f}_2 + \mathcal{O}(\Delta x^2).$$

Furthermore, for the flux values interior to the DG cell, the boundary terms  $[(f^* - f_h)L_j]_{-1}^1$  in (1.63) vanish for Legendre-Gauss-Lobatto nodes and we have

$$f_h(\bar{x}_{j+1}) = \bar{f}_j + \mathcal{O}(\Delta x^2) + \omega_j \sum_{\nu=1}^{N+1} L'_\nu(\xi_j) f_\nu + \mathcal{O}(\Delta x^2) = \bar{f}_{j+1} + \mathcal{O}(\Delta x^2),$$

for  $j = 2, \dots, N$ , without the additional assumptions on the values of the numerical flux function.  $\square$

The reformulation of a DG scheme as a subcell finite volume scheme illuminates the structure of the scheme in the interior of a DG cell. This provides a microscopic view of the DG scheme and allows for direct modifications of the finite volume fluxes  $\bar{f}_j$ . For instance, such modifications are useful for shock capturing as in [179] or in order to recover or create specific split formulations of conservation laws as in [70]. Conversely, the following section offers a macroscopic view on the one-dimensional DG scheme for the linear advection equation using numerical fluxes in the range from upwind to central fluxes. For this purpose, the unknowns on all DG cells are collected into a single global vector and an upwind SBP property is derived for the global DG scheme. This accounts for the fact that the numerical fluxes generally introduce additional numerical dissipation to the DG scheme and thus to the SBP property whereas the cell-wise SBP formulation treats these interface terms separately.

### 1.2.3 The DG scheme with numerical fluxes in upwind SBP framework

According to equation (1.51), the cell-wise DG scheme in matrix-vector formulation applied to the linear advection equation (1.1), with positive advection velocity  $a > 0$ , is given by

$$\frac{\Delta x_i}{2} \frac{d\mathbf{u}^i}{dt} + a\mathbf{D}\mathbf{u}^i = \mathbf{M}^{-1}[(au_h^i - (au)^{*,i})\mathbf{L}]_{-1}^1, \tag{1.64}$$

on interior cells  $I_i$ ,  $i = 2, \dots, K - 1$ . Hereby, a suitable numerical flux function  $(au)^{*,i}$  is given by  $(au)^{*,i}(-1) = (au)_i^*$  and  $(au)^{*,i}(1) = (au)_{i+1}^*$  with

$$(au)_i^* = a \left( \frac{1}{2} + \theta \right) u_h^{i-1}(1) + a \left( \frac{1}{2} - \theta \right) u_h^i(-1), \quad \theta \in \left[ 0, \frac{1}{2} \right]. \quad (1.65)$$

Therefore, the choice of numerical flux functions includes the central flux for  $\theta = 0$  and the upwind flux for  $\theta = \frac{1}{2}$ . Furthermore, similar to the boundary treatment of generalized SBP schemes of the form (1.30), the scheme on boundary cells is given by

$$\frac{\Delta x_1}{2} \frac{d\mathbf{u}^1}{dt} + a\mathbf{D}\mathbf{u}^1 = \mathbf{M}^{-1} (au_h^1(1) - (au)_2^*) \mathbf{L}(1) + \sigma \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T \mathbf{u}^1, \quad (1.66)$$

on the leftmost DG cell  $\Omega_1$ , with SAT parameter  $\sigma$ , and

$$\frac{\Delta x_K}{2} \frac{d\mathbf{u}^K}{dt} + a\mathbf{D}\mathbf{u}^K = -\mathbf{M}^{-1} (au_h^K(-1) - (au)_K^*) \mathbf{L}(-1), \quad (1.67)$$

on the rightmost DG cell  $\Omega_K$  with outgoing information through its right boundary.

The goal of this section is to rewrite the above DG discretization on the complete computational domain as an upwind SBP scheme which includes the interactions of degrees of freedom through adjacent cells. For simplification, periodic boundary conditions are assumed. Under this assumption, we may rewrite Eq. (1.64) as

$$\frac{d\mathbf{u}}{dt} + a\mathbf{D}_{glob}^- \mathbf{u} = \underline{\text{SAT}}, \quad (1.68)$$

with the global nodal DG approximation given by  $\mathbf{u} = (\mathbf{u}^1, \dots, \mathbf{u}^K)^T$  and the extended SAT term  $\underline{\text{SAT}} = (\underline{\text{SAT}}_1^T, 0, \dots, 0)^T$ , where  $\underline{\text{SAT}}_1 = \frac{2\sigma}{\Delta x_1} \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T \mathbf{u}^1$ .

Now, the global DG derivative operator  $\mathbf{D}_{glob}^-$  is a block tridiagonal matrix with blocks corresponding to each DG element and its left and right adjacent cells. More precisely, we have

**Lemma 1.10.** *The global DG derivative operator  $\mathbf{D}_{glob}^- = \mathbf{D}_{glob}^-(\theta)$  in (1.68) is of the block tridiagonal form*

$$\mathbf{D}_{glob}^-(\theta) = \begin{pmatrix} \frac{2}{\Delta x_1} & & & \\ & \frac{2}{\Delta x_2} & & \\ & & \ddots & \\ & & & \frac{2}{\Delta x_K} \end{pmatrix} \begin{pmatrix} \mathbf{A}_{lb}(\theta) & \mathbf{A}_{12}(\theta) & & & \\ \mathbf{A}_{21}(\theta) & \mathbf{A}_{11}(\theta) & \mathbf{A}_{12}(\theta) & & \\ & \ddots & \ddots & \ddots & \\ & & \mathbf{A}_{21}(\theta) & \mathbf{A}_{11}(\theta) & \mathbf{A}_{12}(\theta) \\ & & & \mathbf{A}_{21}(\theta) & \mathbf{A}_{rb}(\theta) \end{pmatrix},$$

where the repeating blocks corresponding to interior cells are given by

$$\begin{aligned} \mathbf{A}_{11}(\theta) &= \mathbf{D} - \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(1)^T + \left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T, \\ \mathbf{A}_{12}(\theta) &= \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(-1)^T, \\ \mathbf{A}_{21}(\theta) &= -\left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(1)^T, \end{aligned}$$

and the boundary treatment yields

$$\begin{aligned}\mathbf{A}_{lb}(\theta) &= \mathbf{D} - \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(1)^T, \\ \mathbf{A}_{rb}(\theta) &= \mathbf{D} + \left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T.\end{aligned}$$

*Proof.* The evaluation of the numerical flux function at an interface given in (1.65) may be rewritten as

$$(au)_i^* = a \left( \frac{1}{2} + \theta \right) \mathbf{L}(1)^T \mathbf{u}^{i-1} + a \left( \frac{1}{2} - \theta \right) \mathbf{L}(-1)^T \mathbf{u}^i$$

For the contribution at the left and right cell boundaries to the DG scheme (1.64), we therefore have

$$\begin{aligned}(au_h^i(-1) - (au)_i^*) \mathbf{L}(-1) &= a \mathbf{L}(-1) \mathbf{L}(-1)^T \mathbf{u}^i - \mathbf{L}(-1)(au)_i^* \\ &= a \left( \frac{1}{2} + \theta \right) \mathbf{L}(-1) \mathbf{L}(-1)^T \mathbf{u}^i - a \left( \frac{1}{2} + \theta \right) \mathbf{L}(-1) \mathbf{L}(1)^T \mathbf{u}^{i-1} \\ &= a \left( \tilde{\mathbf{A}}_1^\theta \mathbf{u}^i + \tilde{\mathbf{A}}_2^\theta \mathbf{u}^{i-1} \right),\end{aligned}$$

and

$$\begin{aligned}- (au_h^i(1) - (au)_{i+1}^*) \mathbf{L}(1) &= -a \mathbf{L}(1) \mathbf{L}(1)^T \mathbf{u}^i + \mathbf{L}(1)(au)_{i+1}^* \\ &= -a \left( \frac{1}{2} - \theta \right) \mathbf{L}(1) \mathbf{L}(1)^T \mathbf{u}^i + a \left( \frac{1}{2} - \theta \right) \mathbf{L}(1) \mathbf{L}(-1)^T \mathbf{u}^{i+1} \\ &= a \left( \tilde{\mathbf{A}}_3^\theta \mathbf{u}^i + \tilde{\mathbf{A}}_4^\theta \mathbf{u}^{i+1} \right).\end{aligned}$$

From (1.64), we therefore derive

$$\begin{aligned}\mathbf{A}_{11}(\theta) &= \mathbf{D} + \mathbf{M}^{-1} \tilde{\mathbf{A}}_3^\theta + \mathbf{M}^{-1} \tilde{\mathbf{A}}_1^\theta \\ &= \mathbf{D} - \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(1)^T + \left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T,\end{aligned}$$

while for the blocks corresponding to the left and right boundary cells, equations (1.66) and (1.67) yield

$$\begin{aligned}\mathbf{A}_{lb}(\theta) &= \mathbf{D} + \mathbf{M}^{-1} \tilde{\mathbf{A}}_3^\theta = \mathbf{D} - \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(1)^T, \\ \mathbf{A}_{rb}(\theta) &= \mathbf{D} + \mathbf{M}^{-1} \tilde{\mathbf{A}}_1^\theta = \mathbf{D} + \left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T,\end{aligned}$$

respectively.

Similarly, the remaining blocks are given by

$$\begin{aligned}\mathbf{A}_{12}(\theta) &= \mathbf{M}^{-1} \tilde{\mathbf{A}}_4^\theta = \left( \frac{1}{2} - \theta \right) \mathbf{M}^{-1} \mathbf{L}(1) \mathbf{L}(1)^T, \\ \mathbf{A}_{21}(\theta) &= \mathbf{M}^{-1} \tilde{\mathbf{A}}_2^\theta = - \left( \frac{1}{2} + \theta \right) \mathbf{M}^{-1} \mathbf{L}(-1) \mathbf{L}(-1)^T.\end{aligned}$$

□

The specific form of  $\mathbf{D}_{glob}^-$  allows us to show that the dual pair  $\{\mathbf{D}^-, \mathbf{D}^+\}$  with  $\mathbf{D}^- = \mathbf{D}_{glob}^-(\theta)$  and a suitably constructed dual operator  $\mathbf{D}^+$  satisfies the upwind SBP properties specified in Definition 1.4.

**Theorem 1.11.** *The dual pair of discrete derivative operators*

$$\mathbf{D}^- = \mathbf{D}_{glob}^-(\theta), \quad \mathbf{D}^+ = \mathbf{D}_{glob}^-(-\theta), \quad (1.69)$$

is a dual pair of diagonal-norm upwind SBP operators with respect to the global diagonal norm

$$\mathbf{M}_{glob} = \text{diag} \left( \frac{\Delta x_1}{2} \mathbf{M}, \dots, \frac{\Delta x_K}{2} \mathbf{M} \right),$$

and the generalized boundary operator

$$\mathbf{B}_{glob} = \text{diag} (\mathbf{B}_l, 0, \dots, 0, \mathbf{B}_r),$$

with  $\mathbf{B}_l = -\mathbf{L}(-1) \mathbf{L}(-1)^T$  and  $\mathbf{B}_r = \mathbf{L}(1) \mathbf{L}(1)^T$ .

*Proof.* The first step is to define the matrices  $\mathbf{Q}^-$  and  $\mathbf{Q}^+$  based on (1.33). Accordingly, we have

$$\begin{aligned}\mathbf{Q}^- &= \mathbf{M}_{glob} \mathbf{D}^- - \frac{1}{2} \mathbf{B}_{glob} = \mathbf{M}_{glob} \mathbf{D}_{glob}^-(\theta) - \frac{1}{2} \mathbf{B}_{glob}, \\ \mathbf{Q}^+ &= \mathbf{M}_{glob} \mathbf{D}^+ - \frac{1}{2} \mathbf{B}_{glob} = \mathbf{M}_{glob} \mathbf{D}_{glob}^-(-\theta) - \frac{1}{2} \mathbf{B}_{glob}.\end{aligned}$$

Next, we show that  $\mathbf{Q}^-$  and  $\mathbf{Q}^+$  satisfy the SBP property (1.34), i.e. that the dual operator  $\mathbf{D}^+$  is chosen properly. We have

$$\begin{aligned}\mathbf{Q}^+ + (\mathbf{Q}^-)^T &= \mathbf{M}_{glob} \mathbf{D}_{glob}^-(-\theta) + (\mathbf{D}_{glob}^-(\theta))^T \mathbf{M}_{glob} - \mathbf{B}_{glob} \\ &= \begin{pmatrix} \mathbf{Q}_{lb} & \mathbf{Q}_{12} & & \\ \mathbf{Q}_{21} & \mathbf{Q}_{11} & \mathbf{Q}_{12} & \\ & \ddots & \ddots & \ddots \\ & & \mathbf{Q}_{21} & \mathbf{Q}_{11} & \mathbf{Q}_{12} \\ & & & \mathbf{Q}_{21} & \mathbf{Q}_{rb} \end{pmatrix},\end{aligned}$$

with

$$\begin{aligned}
\mathbf{Q}_{11} &= \mathbf{M} \mathbf{A}_{11}(-\theta) + \mathbf{A}_{11}^T(\theta) \mathbf{M} = \mathbf{M} \mathbf{D} + \mathbf{D}^T \mathbf{M} - \mathbf{L}(1) \mathbf{L}(1)^T + \mathbf{L}(-1) \mathbf{L}(-1)^T \\
&= \mathbf{M} \mathbf{D} + \mathbf{D}^T \mathbf{M} - \mathbf{B} = \mathbf{0}, \\
\mathbf{Q}_{lb} &= \mathbf{M} \mathbf{A}_{lb}(-\theta) + \mathbf{A}_{lb}^T(\theta) \mathbf{M} - \mathbf{B}_l = \mathbf{M} \mathbf{D} + \mathbf{D}^T \mathbf{M} - \mathbf{L}(1) \mathbf{L}(1)^T + \mathbf{L}(-1) \mathbf{L}(-1)^T = \mathbf{0}, \\
\mathbf{Q}_{rb} &= \mathbf{M} \mathbf{A}_{rb}(-\theta) + \mathbf{A}_{rb}^T(\theta) \mathbf{M} - \mathbf{B}_r = \mathbf{M} \mathbf{D} + \mathbf{D}^T \mathbf{M} + \mathbf{L}(-1) \mathbf{L}(-1)^T - \mathbf{L}(1) \mathbf{L}(1)^T = \mathbf{0}, \\
\mathbf{Q}_{12} &= \mathbf{M} \mathbf{A}_{12}(-\theta) + \mathbf{A}_{21}^T(\theta) \mathbf{M} = \left( \frac{1}{2} + \theta \right) \mathbf{L}(1) \mathbf{L}(-1)^T - \left( \frac{1}{2} + \theta \right) \mathbf{L}(1) \mathbf{L}(-1)^T = \mathbf{0}, \\
\mathbf{Q}_{21} &= \mathbf{M} \mathbf{A}_{21}(-\theta) + \mathbf{A}_{12}^T(\theta) \mathbf{M} = - \left( \frac{1}{2} - \theta \right) \mathbf{L}(-1) \mathbf{L}(1)^T + \left( \frac{1}{2} - \theta \right) \mathbf{L}(-1) \mathbf{L}(1)^T = \mathbf{0}.
\end{aligned}$$

Therefore, we have the desired result  $\mathbf{Q}^+ + (\mathbf{Q}^-)^T = \mathbf{0}$ .

It remains to show the stability constraint (1.35), i.e. to show that

$$\mathbf{C} = \frac{1}{2} (\mathbf{Q}^+ - \mathbf{Q}^-) = \frac{1}{2} \mathbf{M}_{glob} \left( \mathbf{D}_{glob}^-(-\theta) - \mathbf{D}_{glob}^+(\theta) \right) \quad (1.70)$$

is negative semi-definite. We have

$$\mathbf{C} = \begin{pmatrix} \mathbf{C}_{lb} & \mathbf{C}_{12} & & \\ \mathbf{C}_{21} & \mathbf{C}_{11} & \mathbf{C}_{12} & \\ & \ddots & \ddots & \ddots \\ & & \mathbf{C}_{21} & \mathbf{C}_{11} & \mathbf{C}_{12} \\ & & & \mathbf{C}_{21} & \mathbf{C}_{rb} \end{pmatrix},$$

with

$$\begin{aligned}
\mathbf{C}_{11} &= \frac{1}{2} (\mathbf{M} \mathbf{A}_{11}(-\theta) - \mathbf{M} \mathbf{A}_{11}(\theta)) = -\theta (\mathbf{L}(1) \mathbf{L}(1)^T + \mathbf{L}(-1) \mathbf{L}(-1)^T), \\
\mathbf{C}_{lb} &= \frac{1}{2} (\mathbf{M} \mathbf{A}_{lb}(-\theta) - \mathbf{M} \mathbf{A}_{lb}(\theta)) = -\theta \mathbf{L}(1) \mathbf{L}(1)^T, \\
\mathbf{C}_{rb} &= \frac{1}{2} (\mathbf{M} \mathbf{A}_{rb}(-\theta) - \mathbf{M} \mathbf{A}_{rb}(\theta)) = -\theta \mathbf{L}(-1) \mathbf{L}(-1)^T, \\
\mathbf{C}_{12} &= \frac{1}{2} (\mathbf{M} \mathbf{A}_{12}(-\theta) - \mathbf{M} \mathbf{A}_{12}(\theta)) = \theta \mathbf{L}(1) \mathbf{L}(-1)^T, \\
\mathbf{C}_{21} &= \frac{1}{2} (\mathbf{M} \mathbf{A}_{21}(-\theta) - \mathbf{M} \mathbf{A}_{21}(\theta)) = \theta \mathbf{L}(-1) \mathbf{L}(1)^T = \mathbf{C}_{12}^T.
\end{aligned}$$

Hence, we have

$$\begin{aligned}
\mathbf{u}^T \mathbf{C} \mathbf{u} &= (\mathbf{u}^1)^T \mathbf{C}_{lb} \mathbf{u}^1 + \sum_{i=2}^{K-1} (\mathbf{u}^i)^T \mathbf{C}_{11} \mathbf{u}^i + (\mathbf{u}^K)^T \mathbf{C}_{rb} \mathbf{u}^K + 2 \sum_{i=1}^{K-1} (\mathbf{u}^i)^T \mathbf{C}_{12} \mathbf{u}^{i+1} \\
&= -\theta \left( (u_h^1(1))^2 + \sum_{i=2}^{K-1} ((u_h^i(-1))^2 + (u_h^i(1))^2) + (u_h^K(-1))^2 - 2 \sum_{i=1}^{K-1} u_h^i(1) u_h^{i+1}(-1) \right) \\
&= -\theta \sum_{i=1}^{K-1} (u_h^{i+1}(-1) - u_h^i(1))^2 \leq 0,
\end{aligned}$$

i.e.  $\mathbf{C}$  is negative semi-definite.  $\square$

The following two remarks regarding the DG scheme viewed as generalized upwind SBP scheme are in order.

**Remark 1.12.** Since  $\mathbf{D}_{glob}^+(\theta) = \mathbf{D}_{glob}^-(-\theta)$ , we note that

1. the dual operator  $\mathbf{D}_{glob}^+$  arises naturally from the DG scheme applied to the linear advection equation with negative advection velocity  $a < 0$ , and
2. for  $\theta = 0$ , the global DG operator  $\mathbf{D}_{glob} = \mathbf{D}_{glob}^-(0) = \mathbf{D}_{glob}^+(0)$  is a generalized diagonal-norm SBP operator in the sense of Definition 1.3.

The above generalized upwind SBP operators resulting from the DG discretization in one space dimension may also be used to construct second-derivative operators as discussed in the framework of finite difference SBP schemes in Section 1.1. In particular, the application of (1.39) to the operators  $\mathbf{D}_{glob}^-(\theta)$  and  $\mathbf{D}_{glob}^+(\theta)$  will be related to classical DG diffusion fluxes in Section 2.1.

#### 1.2.4 Extension to 2D cartesian grids via tensor-product SBP operators

In this section as well as Section 1.2.5, we consider two-dimensional scalar hyperbolic conservation laws of the form

$$\frac{\partial}{\partial t} u(\mathbf{x}, t) + \nabla \cdot \mathbf{f}(u(\mathbf{x}, t)) = 0, \quad (\mathbf{x}, t) \in \Omega \times \mathbb{R}_+, \quad (1.71)$$

where  $\Omega \subset \mathbb{R}^2$  is an open polygonal domain in which initial conditions  $u(\mathbf{x}, 0) = u_0(\mathbf{x})$  are given. In the following, the components of the flux vector  $\mathbf{f}$  will be denoted by  $\mathbf{f} = (f^x, f^y)^T$ , consistent with the notation  $\mathbf{x} = (x, y)^T$  for a point in the two-dimensional computational domain  $\Omega$ . Furthermore, we assume appropriately posed boundary conditions.

The SBP properties of DG schemes in one space dimension easily extend to two-dimensional cartesian grids if tensor-product basis functions  $L_i(\xi)L_j(\eta)$  are used on the reference element  $K = [-1, 1]^2$ . In this case, the DG scheme in matrix-vector formulation can be constructed based on the 1D formulation by using Kronecker products. This 2D extension of the SBP properties of nodal DG schemes will be derived in this section. In Section 1.4.3, this SBP

property on tensor-product grids will be related to the favorable quality of a kinetic energy preserving DG scheme applied in the context of two-dimensional turbulent flow.

In the following, the presentation will be restricted to the nodal DG scheme on Legendre-Gauss quadrature nodes. For given pointwise data values  $g_{i,j} \approx g(\xi_i, \eta_j)$  at the two-dimensional Legendre-Gauss nodes  $\xi_i, \eta_j$ ,  $i, j = 1, \dots, N+1$ , the vector  $\mathbf{g}$  collects these values in the form

$$g_{\nu(i,j)} = g_{(i-1)(N+1)+j} = g_{(i,j)},$$

i.e. in lexicographical order. Denoting by  $\mathbf{M}_{1D}$  the mass matrix of the DG scheme in one space dimension and by  $\mathbf{I}_{1D}$  the corresponding identity matrix whilst keeping the definition of  $\mathbf{D}$  and  $\mathbf{S}$  as in Section 1.2.1 the DG scheme in two space dimensions then uses the mass, stiffness and differentiation matrices

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_{1D} \otimes \mathbf{M}_{1D}, \\ \mathbf{S}_\xi &= \mathbf{S} \otimes \mathbf{M}_{1D}, & \mathbf{S}_\eta &= \mathbf{M}_{1D} \otimes \mathbf{S}_\eta, \\ \mathbf{D}_\xi &= \mathbf{D} \otimes \mathbf{I}_{1D} = \mathbf{M}^{-1} \mathbf{S}_\xi, & \mathbf{D}_\eta &= \mathbf{I}_{1D} \otimes \mathbf{D} = \mathbf{M}^{-1} \mathbf{S}_\eta, \end{aligned}$$

as well as the boundary operators

$$\mathbf{B}_\xi = \mathbf{B} \otimes \mathbf{M}_{1D}, \quad \mathbf{B}_\eta = \mathbf{M}_{1D} \otimes \mathbf{B}.$$

Furthermore, using the properties of the Kronecker product, we may easily derive the SBP properties  $\mathbf{S}_\xi + \mathbf{S}_\xi^T = \mathbf{B}_\xi$  and  $\mathbf{S}_\eta + \mathbf{S}_\eta^T = \mathbf{B}_\eta$ .

The action of the boundary operators on pointwise values is related to the discrete boundary integral as follows. Let  $\omega_k$  denote the Legendre-Gauss weights as given in Section 1.2.1. Furthermore, we will introduce the index  $e$  enumerating the edges of the reference square  $K$  in counter-clockwise manner starting with  $e = 1$  referring to the lower edge. The corresponding normal vectors are denoted by  $\mathbf{n}^e = (n_\xi^e, n_\eta^e)$ , i.e.  $n_\xi^1 = 0, n_\eta^1 = -1$  and the nodes  $(\xi_k^e, \eta_k^e)$  denote the Legendre-Gauss quadrature nodes on edge  $e$ . The following Lemma then transfers the actions of  $\mathbf{B}_\xi$  and  $\mathbf{B}_\eta$  to a numerical quadrature on  $\partial K$ .

**Lemma 1.13.** *Let  $\mathbf{g}^\xi, \mathbf{g}^\eta \in \mathbb{R}^{(N+1)^2}$  denote arbitrary sets of pointwise data values. For the sum of boundary terms  $\mathbf{B}_\xi \mathbf{g}^\xi + \mathbf{B}_\eta \mathbf{g}^\eta$  we then have*

$$\mathbf{B}_\xi \mathbf{g}^\xi + \mathbf{B}_\eta \mathbf{g}^\eta = \sum_{e=1}^4 \sum_{k=1}^{N+1} \omega_k \left( n_\xi^e g_h^\xi(\xi_k^e, \eta_k^e) + n_\eta^e g_h^\eta(\xi_k^e, \eta_k^e) \right) \mathbf{L}(\xi_k^e, \eta_k^e),$$

where  $\mathbf{L}(\xi_k^e, \eta_k^e) = \mathbf{L}(\xi_k^e) \otimes \mathbf{L}(\eta_k^e)$  and  $g_h^\xi$  and  $g_h^\eta$  denote the polynomial interpolations

$$g_h^\xi(\xi, \eta) = \sum_{i=1}^{N+1} \sum_{j=1}^{N+1} g_{\nu(i,j)}^\xi L_i(\xi) L_j(\eta), \quad g_h^\eta(\xi, \eta) = \sum_{i=1}^{N+1} \sum_{j=1}^{N+1} g_{\nu(i,j)}^\eta L_i(\xi) L_j(\eta).$$

*Proof.* The columns of the matrices  $\mathbf{B}_\xi = \mathbf{B} \otimes \mathbf{M}_{1D}$  and  $\mathbf{B}_\eta = \mathbf{M}_{1D} \otimes \mathbf{B}$  can be related to the Lagrange polynomials as follows. Let  $\nu(i, j) = (i-1)(N+1) + j$ . From Theorem 1.5

in Section 1.2.1, we recall that  $\mathbf{B}$  has the entries  $B_{jk} = [L_j L_k]_{-1}^1$  and  $\mathbf{M}_{1D}$  is diagonal with entries  $\omega_j$ . Therefore, the  $\nu$ -th columns of the above matrices  $\mathbf{B}_\xi$  and  $\mathbf{B}_\eta$  are given by  $\mathbf{b}_{\xi \nu(i,j)} = [\mathbf{L}(\xi) L_i(\xi)]_{-1}^1 \otimes \omega_j \epsilon_j$  and  $\mathbf{b}_{\eta \nu(i,j)} = \omega_i \epsilon_i \otimes [\mathbf{L}(\xi) L_j(\xi)]_{-1}^1$ , respectively, where  $\epsilon_j$  denotes the  $j$ -th unit vector.

Therefore, we obtain

$$\begin{aligned} \mathbf{B}_\xi \mathbf{g}^\xi &= \sum_{i=1}^{N+1} \sum_{j=1}^{N+1} \mathbf{b}_{\xi \nu(i,j)} g_{\nu(i,j)}^\xi = \sum_{i=1}^{N+1} \sum_{j=1}^{N+1} ([\mathbf{L}(\xi) L_i(\xi)]_{-1}^1 \otimes \omega_j \epsilon_j) g_{\nu(i,j)}^\xi \\ &= \left[ \sum_{j=1}^{N+1} \omega_j \left( \sum_{i=1}^{N+1} L_i(\xi) g_{\nu(i,j)}^\xi \right) \mathbf{L}(\xi) \otimes \epsilon_j \right]_{-1}^1 = \left[ \sum_{j=1}^{N+1} \omega_j g_h^\xi(\xi, \eta_j) \mathbf{L}(\xi) \otimes \mathbf{L}(\eta_j) \right]_{-1}^1. \end{aligned}$$

Since for the normal vectors and Legendre-Gauss nodes on edges  $e = 2$  and  $e = 4$  of the reference square we have

$$n_\xi^2 = 1, n_\xi^4 = -1 \quad \text{and} \quad \xi_j^2 = 1, \xi_j^4 = -1, j = 1, \dots, N+1,$$

it holds that

$$\mathbf{B}_\xi \mathbf{g}^\xi = \sum_{e=2,4} \sum_{j=1}^{N+1} n_\xi^e \omega_j g_h^\xi(\xi_j^e, \eta_j^e) \mathbf{L}(\xi_j^e) \otimes \mathbf{L}(\eta_j^e).$$

With analogous arguments we obtain

$$\mathbf{B}_\eta \mathbf{g}^\eta = \left[ \sum_{i=1}^{N+1} \omega_i \epsilon_i \otimes \mathbf{L}(\eta) \sum_{j=1}^{N+1} L_j(\eta) g_{\nu(i,j)}^\eta \right]_{-1}^1 = \sum_{e=1,3} \sum_{i=1}^{N+1} n_\eta^e \omega_i g_h^\eta(\xi_i^e, \eta_i^e) \mathbf{L}(\xi_i^e) \otimes \mathbf{L}(\eta_i^e).$$

Summing up  $\mathbf{B}_\xi \mathbf{g}^\xi + \mathbf{B}_\eta \mathbf{g}^\eta$  and considering that  $n_\xi^1 = n_\eta^2 = n_\xi^3 = n_\eta^4 = 0$  we obtain the assertion.  $\square$

**Remark 1.14.** To reduce formalism in the later definition (1.73) of the DG scheme in two space dimensions on tensor-product grids, the short notation

$$\langle \Psi \rangle_{\partial K} = \sum_{e=1}^4 \sum_{k=1}^{N+1} \omega_k \Psi(\xi_k^e, \eta_k^e)$$

will be used to denote surface terms of the above form, i.e. we have

$$\mathbf{B}_\xi \mathbf{g}^\xi + \mathbf{B}_\eta \mathbf{g}^\eta = \left\langle \left( n_\xi g_h^\xi + n_\eta g_h^\eta \right) \mathbf{L}(\xi, \eta) \right\rangle_{\partial K}.$$

With this notation, the extension of the standard DG scheme in weak form to the two-dimensional conservation law (1.71) with flux vector  $\mathbf{f} = (f^x, f^y)^T$  on a cartesian grid is given by the cell-wise formulation

$$\frac{\Delta x \Delta y}{4} \mathbf{M} \frac{d\mathbf{u}}{dt} - \mathbf{S}_\xi^T \mathbf{f}^\xi - \mathbf{S}_\eta^T \mathbf{f}^\eta = - \langle f^* \mathbf{L}(\xi, \eta) \rangle_{\partial K}, \quad (1.72)$$

where  $\mathbf{f}^\xi$  and  $\mathbf{f}^\eta$  approximate the grid values of the fluxes  $f^x$  and  $f^y$ , respectively, on the specific grid cell with length scales  $\Delta x, \Delta y$ . Furthermore, the right-hand side of (1.72) contains the numerical flux function  $f^*$  and again employs the short notation of Remark 1.14 such that  $f^*$  is in fact evaluated at the Legendre-Gauss quadrature nodes on each edge. More precisely, the numerical flux function depends on three arguments which are the corresponding one-sided limits  $u_L$  and  $u_R$  of the DG solution from the interior and exterior sides of the interface, respectively, and the normal vector  $\mathbf{n}^e$  of the respective edge. Therefore, we have

$$-\langle f^* \mathbf{L}(\xi, \eta) \rangle_{\partial K} = -\sum_{e=1}^4 \sum_{k=1}^{N+1} \omega_k f^*(u_L(\xi_k^e, \eta_k^e), u_R(\xi_k^e, \eta_k^e), \mathbf{n}^e) \mathbf{L}(\xi_k^e, \eta_k^e).$$

The strong form corresponding to (1.72) is obtained by multiplication with  $\mathbf{M}^{-1}$  and application of Lemma 1.13, i.e.

$$\frac{\Delta x \Delta y}{4} \frac{d\mathbf{u}}{dt} + \mathbf{D}_\xi \mathbf{f}^\xi + \mathbf{D}_\eta \mathbf{f}^\eta = \mathbf{M}^{-1} \left\langle \left( n_\xi f_h^\xi + n_\eta f_h^\eta - f^* \right) \mathbf{L}(\xi, \eta) \right\rangle_{\partial K}. \quad (1.73)$$

### 1.2.5 Extension to DG schemes on triangular grids

In this section, we consider the discretization of the two-dimensional scalar hyperbolic conservation law (1.71) on unstructured triangular grids.

Let  $\mathcal{T}^h$  be a conforming triangulation of the closure  $\bar{\Omega}$  of the computational domain and let  $V^h$  be the piecewise polynomial space defined by  $V^h = \{v_h \in L^\infty(\Omega) \mid v_h|_{\tau_i} \in P^N(\tau_i) \quad \forall \tau_i \in \mathcal{T}^h\}$ , where  $P^N(\tau_i)$  denotes the space of all polynomials on  $\tau_i$  of degree  $\leq N$ . Multiplying equation (1.71) by test functions in  $V^h$ , integrating over  $\Omega$  and using the divergence theorem leads to the semi-discrete equation

$$\frac{d}{dt} \int_{\Omega} u v_h d\mathbf{x} + \sum_{\tau_i \in \mathcal{T}^h} \left( \int_{\partial \tau_i} \mathbf{f}(u) \cdot \mathbf{n} v_h d\sigma - \int_{\tau_i} \mathbf{f}(u) \cdot \nabla v_h d\mathbf{x} \right) = 0, \quad \forall v_h \in V^h \quad (1.74)$$

As (1.74) is linear in  $v_h$ , for each triangular subset  $\tau_i \in \mathcal{T}^h$  it is sufficient to consider only those test functions  $v_h$  vanishing outside  $\tau_i$  and we obtain the local semi-discrete equation

$$\frac{d}{dt} \int_{\tau_i} u \Phi d\mathbf{x} + \int_{\partial \tau_i} \mathbf{f}(u) \cdot \mathbf{n} \Phi d\sigma - \int_{\tau_i} \mathbf{f}(u) \cdot \nabla \Phi d\mathbf{x} = 0, \quad (1.75)$$

valid for any triangular subset  $\tau_i \in \mathcal{T}^h$  and any polynomial  $\Phi \in P^N(\tau_i)$ .

Analogously to the one-dimensional case, the DG discretization in space then constructs an approximation

$$u_h : \Omega \times \mathbb{R}_+ \rightarrow \mathbb{R}$$

of  $u$  with  $u_h(\cdot, t) \in V^h$  for  $t \in \mathbb{R}_+^0$ . The approximation  $u_h$  is furthermore supposed to satisfy an equation similar to (1.75). As, in general,  $u_h \in V^h$  is discontinuous at the cell interfaces, we cannot simply substitute  $u_h$  for  $u$  in (1.75). Indeed, if the term  $\mathbf{f}(u) \cdot \mathbf{n}$  is just replaced by  $\mathbf{f}(u_h) \cdot \mathbf{n}$  the triangular elements are completely decoupled, leading to a physically unreasonable scheme. To this end, we again employ a numerical flux function  $f^*$  which depends on the

physical quantities  $u_L$  and  $u_R$  as well as the normal vector  $\mathbf{n}$ . The inclusion of the numerical flux function  $f^*$  leads to the following space discretization,

$$\frac{d}{dt} \int_{\tau_i} u_h \Phi d\mathbf{x} + \sum_{j=1}^3 \int_{\Gamma_{ij}} f^*(u_h^i, u_h^{ij}, \mathbf{n}_{ij}) \Phi d\sigma - \int_{\tau_i} \mathbf{f}(u_h) \cdot \nabla \Phi d\mathbf{x} = 0, \quad (1.76)$$

where  $\partial\tau_i$  is decomposed into straight edges  $\Gamma_{ij}$ , i.e.  $\partial\tau_i = \cup_{j=1}^3 \Gamma_{ij}$ , with length  $|\Gamma_{ij}|$  and normal vector  $\mathbf{n}_{ij}$  and where we incorporated the interface and boundary treatment using the numerical flux function  $f^*$  in the following way. To denote adjacent elements, if there is an element  $\tau_k \in \mathcal{T}^h$  with  $\Gamma_{ij} = \partial\tau_i \cap \partial\tau_k$  we denote the index  $k$  by  $n(i,j)$ . For the boundary treatment we decompose  $\partial\Omega$  into an inflow and an outflow boundary part as  $\partial\Omega = \Gamma_{in} \cup \Gamma_{out}$ . Other types of boundary treatment, e.g. wall boundary conditions, are ignored at this point. We then employ the short notation  $u_h^i = u_h|_{\tau_i}$  and incorporate the boundary conditions into  $u_h^{ij}$  by setting

$$u_h^{ij} = \begin{cases} u_h^{n(i,j)} & \text{if } \Gamma_{ij} = \partial\tau_i \cap \partial\tau_{n(i,j)}, \\ u|_{\Gamma_{ij}} & \text{if } \Gamma_{ij} \subset \Gamma_{in}, \\ u_h^i & \text{if } \Gamma_{ij} \subset \Gamma_{out}. \end{cases}$$

### Specific choices of DG basis functions

The approximation  $u_h(\cdot, t)$  can be represented by time-dependent coefficients corresponding to a suitable basis of  $P^N(\tau_i)$ , thus comprising the spacial degrees of freedom of the method. If the specified basis consists of hierarchical orthogonal functions, one may speak of a *modal* DG method, whereas choosing a Lagrange basis for a certain set of interpolation nodes leads to a *nodal* DG scheme. The modal DG scheme constructed and investigated in [134, 135], for instance, is based on the so called PKD polynomials originally constructed by Dubiner [51] for spectral methods on triangular grids. The PKD polynomials are orthogonal polynomials on a reference triangle  $\mathbb{T}$  constructed using one-dimensional Jacobi polynomials  $P_n^{\alpha, \beta}$ . These two-dimensional polynomials as well as their properties and use for spectral methods are also described in detail in [97].

On the reference element  $\mathbb{T} = \{(r, s) \in \mathbb{R}^2 \mid -1 \leq r, s; r + s \leq 0\}$  the PKD polynomials are given by

$$\Phi_{lm}(r, s) = P_l^{0,0} \left( 2 \frac{1+r}{1-s} - 1 \right) \left( \frac{1-s}{2} \right)^l P_m^{2l+1,0}(s), \quad l, m \in \mathbb{N}_0. \quad (1.77)$$

The set  $\{\Phi_{lm} \mid 0 \leq l + m \leq N\}$  collecting the PKD polynomials of maximum degree  $l + m$  represents an orthogonal basis of  $P^N(\mathbb{T})$  with  $\gamma_{lm} := \|\Phi_{lm}\|_{L^2(\mathbb{T})}^2 = \frac{2}{(2l+1)(l+m+1)}$ . Due to their construction, they are also called a *warped tensor product* basis.

Denoting by

$$\psi_i : \tau_i \rightarrow \mathbb{T}, \quad \mathbf{x} \mapsto \mathbf{A}_i \mathbf{x} + \mathbf{b}_i, \quad \mathbf{A}_i \in \mathbb{R}^{2 \times 2}, \quad \mathbf{b}_i \in \mathbb{R}^2, \quad (1.78)$$

an orientation-preserving affine transformation which maps the specific triangle  $\tau_i$  to the reference element  $\mathbb{T}$ , we obtain a basis of  $P^N(\tau_i)$  consisting of the polynomials  $\Phi_{lm} \circ \psi_i$ ,  $0 \leq$

$l + m \leq N$ . Thus, the approximate solution  $u_h|_{\tau_i}$  on a specific triangle can be expanded as

$$u_h(\psi_i^{-1}(r, s), t) = \sum_{l+m \leq N} \hat{u}_{lm}^i(t) \Phi_{lm}(r, s),$$

where the time-dependent functions  $\hat{u}_{lm}^i$  are called the PKD coefficients. Exploiting orthogonality now yields

$$\hat{u}_{lm}^i(t) = \frac{1}{\gamma_{lm}|\tau_i|} \int_{\mathbb{T}} u_h(\psi_i^{-1}(r, s), t) \Phi_{lm}(r, s) dr ds. \quad (1.79)$$

Therefore, equation (1.75) written in the PKD coefficients gives

$$\frac{d}{dt} \hat{u}_{lm}^i = -\frac{2}{\gamma_{lm}|\tau_i|} \int_{\partial\tau_i} \mathbf{f}(u_h) \cdot \mathbf{n} (\Phi_{lm} \circ \psi_i) d\sigma + \frac{1}{\gamma_{lm}} \int_{\mathbb{T}} \mathbf{f}(u_h \circ \psi_i^{-1}) \cdot \mathbf{A}_i^T \nabla_{r,s} \Phi_{lm} dr ds, \quad (1.80)$$

for  $0 \leq l, m \leq N$ , where we employed the transformation of spatial derivatives to the reference element  $\nabla = \nabla_{\mathbf{x}} = \mathbf{A}_i^T \nabla_{r,s}$  as well as the fact that the determinant of the Jacobian  $A_i^{-1}$  of the map  $\psi_i^{-1}$  is equal to  $|\tau_i|/2$ , denoting by  $|\tau_i|$  the area of the triangle  $\tau_i$ .

### Numerical quadrature for the DG scheme on triangular grids

We now approximate the integrals in (1.76) by quadrature formulae which shall be exact for polynomials of degree  $2N$  in the reference element and for polynomials of degree  $2N + 1$  on each part of the cell boundaries  $\partial\tau_i$  respectively, in order to obtain a truncation error of order  $N + 1$  in (1.80), see [42]. To this end, we employ Legendre-Gauss quadrature at the interelement and domain boundaries while a high order quadrature rule for the elements is constructed similar to [97] by using a singular transformation of the reference triangle to the square  $[-1, 1]^2$  and one-dimensional Legendre-Gauss quadrature rules. The resulting quadrature points for the volume integral are completely located within the interior of  $\mathbb{T}$ . Due to the connection to classical tensor-product quadrature rules, this specific construction of a high order quadrature rule on a triangle is sometimes referred to as *warped tensor-product quadrature rule*.

In order to include the quadrature rules and the numerical flux function into the semi-discrete system, we need to introduce the following notation. Let  $\xi_\nu \in [-1, 1]$ ,  $\nu = 1, \dots, n_{edge}$  denote the Legendre-Gauss integration points with corresponding weights  $\omega_\nu$  and let  $(r_\mu, s_\mu) \in \mathbb{T}$ , with  $\mu = 1, \dots, n_{inner}$  be the integration points on  $\mathbb{T}$  with associated weights  $\tilde{\omega}_\mu$ . Furthermore, let  $\mathbf{x}_{ij} : [-1, 1] \rightarrow \Gamma_{ij}$  be the affine transformation mapping the Legendre-Gauss points from  $[-1, 1]$  to  $\Gamma_{ij}$ . The semi-discrete system for the coefficients  $\hat{u}_{lm}^i$  is then given by

$$\begin{aligned} \frac{d}{dt} \hat{u}_{lm}^i &= -\frac{1}{\gamma_{lm}|\tau_i|} \sum_{j=1}^3 |\Gamma_{ij}| \sum_{\nu=1}^{n_{edge}} \omega_\nu f^* \left( u_h^i(\mathbf{x}_{ij}(\xi_\nu), t), u_h^{ij}(\mathbf{x}_{ij}(\xi_\nu), t), \mathbf{n}_{ij} \right) \Phi_{lm}(\psi_i(\mathbf{x}_{ij}(\xi_\nu))) \\ &+ \frac{1}{\gamma_{lm}} \sum_{\mu=1}^{n_{inner}} \tilde{\omega}_\mu \mathbf{f}(u_h \circ \psi_i^{-1}(r_\mu, s_\mu)) \cdot \mathbf{A}_i^T \nabla_{r,s} \Phi_{lm}(r_\mu, s_\mu). \end{aligned} \quad (1.81)$$

On the other hand, the variational formulation (1.76) supplemented by the above quadrature formulae is modified to the semi-discretization

$$\begin{aligned} \frac{d}{dt} \int_{\tau_i} u_h \Phi d\mathbf{x} &+ \sum_{j=1}^3 \frac{|\Gamma_{ij}|}{2} \sum_{\nu=1}^{n_{edge}} \omega_\nu f^* \left( u_h^i(\mathbf{x}_{ij}(\xi_\nu), t), u_h^{ij}(\mathbf{x}_{ij}(\xi_\nu), t), \mathbf{n}_{ij} \right) \Phi(\psi_i(\mathbf{x}_{ij}(\xi_\nu))) \\ &- \frac{|\tau_i|}{2} \sum_{\mu=1}^{n_{inner}} \tilde{\omega}_\mu \mathbf{f}(u_h \circ \psi_i^{-1}(r_\mu, s_\mu)) \cdot \nabla \Phi(\psi_i^{-1}(r_\mu, s_\mu)) d\mathbf{x} = 0, \end{aligned} \quad (1.82)$$

to be satisfied for any  $\Phi \in P^N(\tau_i)$ .

### An SBP scheme based on a nodal version of the DG method

We may rewrite the modal DG scheme (1.82) using warped tensor-product quadrature rules in a nodal version as follows.

Let  $u_j = u_h^i(\mathbf{x}_j)$  denote pointwise values of the approximate solution at specified interpolation points  $\mathbf{x}_1, \dots, \mathbf{x}_{N_I}$  in a given triangle  $\tau_i$ . These points have to be chosen depending on the order of the DG scheme, i.e. the number of interpolation points  $N_I$  equals the number of modal basis functions, which is precisely the number of degrees of freedom within each triangle.

A reasonable set of interpolation points for the definition of a nodal DG scheme on triangular grids is the set of Blyth-Pozrikidis points specified in [20]. Using the Lagrange polynomials  $L_j$  corresponding to the nodes  $\mathbf{x}_j$ , i.e.  $L_j(\mathbf{x}_k) = \delta_{jk}$ , the approximate solution  $u_h^i$  can then be expanded as  $u_h^i(\mathbf{x}) = \sum_j u_j L_j(\mathbf{x})$  and we define the vector of basis functions  $\mathbf{L} = (L_1, \dots, L_{N_I})^T$ .

Collecting the nodal values of  $u_h^i$  with respect to the Lagrange basis  $\mathbf{L}$  in  $\mathbf{u} = (u_1, \dots, u_{N_I})^T \in \mathbb{R}^{N_I}$  and the flux values at the quadrature nodes  $\psi_i^{-1}(r_\mu, s_\mu)$ ,  $\mu = 1, \dots, n_{inner}$  in

$$\tilde{\mathbf{f}}_l = (f_l(u_h \circ \psi_i^{-1}(r_1, s_1)), \dots, f_l(u_h \circ \psi_i^{-1}(r_{n_{inner}}, s_{n_{inner}})))^T,$$

where  $l = 1, 2$ , the modal DG scheme (1.82) can be rewritten in a nodal version as

$$\mathbf{M} \frac{d}{dt} \mathbf{u} - \tilde{\mathbf{D}}_x^T \mathbf{W} \tilde{\mathbf{f}}_1 - \tilde{\mathbf{D}}_y^T \mathbf{W} \tilde{\mathbf{f}}_2 = - \langle f^*(u_h^-, u_h^+, \mathbf{n}) \mathbf{L} \rangle_{\partial \tau_i}. \quad (1.83)$$

Hereby, equation (1.83) deserves some explanation which is given in the following. Firstly, the entries of the mass matrix  $\mathbf{M} \in \mathbb{R}^{N_I \times N_I}$  are given by

$$M_{jk} = \int_{\tau_i} L_j(\mathbf{x}) L_k(\mathbf{x}) d\mathbf{x} = M_{kj}, \quad (1.84)$$

since inserting  $\Phi = L_k$  into the first integral in (1.82) yields

$$\begin{aligned} \frac{d}{dt} \int_{\tau_i} u_h(\mathbf{x}) L_k(\mathbf{x}) d\mathbf{x} &= \frac{d}{dt} \int_{\tau_i} \sum_j u_j L_j(\mathbf{x}) L_k(\mathbf{x}) d\mathbf{x} = \frac{d}{dt} \sum_j u_j(t) \int_{\tau_i} L_j(\mathbf{x}) L_k(\mathbf{x}) d\mathbf{x} \\ &= \sum_j M_{jk} \frac{d}{dt} u_j(t). \end{aligned}$$

Secondly, the third term on the left-hand side of (1.82) is rewritten by inserting the quadrature weights into the diagonal matrix  $\mathbf{W} \in \mathbb{R}^{n_{inner} \times n_{inner}}$ , i.e.  $\mathbf{W} = \frac{|\tau_i|}{2} \text{diag}(\tilde{\omega}_1, \dots, \tilde{\omega}_{n_{inner}})$ , and defining the differentiation matrices  $\tilde{\mathbf{D}}_x, \tilde{\mathbf{D}}_y \in \mathbb{R}^{n_{inner} \times N_I}$  with entries  $\tilde{D}_{x,\mu k} = \frac{\partial}{\partial x} L_k(\psi_i^{-1}(r_\mu, s_\mu))$  and  $\tilde{D}_{y,\mu k} = \frac{\partial}{\partial y} L_k(\psi_i^{-1}(r_\mu, s_\mu))$ , respectively. Furthermore, analogously to Remark 1.14, the interface quadrature terms in (1.82) are abbreviated as

$$\langle f^*(u_h^-, u_h^+, \mathbf{n}) \mathbf{L} \rangle_{\partial\tau_i} = \sum_{j=1}^3 \frac{|\Gamma_{ij}|}{2} \sum_{\nu=1}^{n_{edge}} \omega_\nu f^* \left( u_h^i(\mathbf{x}_{ij}(\xi_\nu), t), u_h^{ij}(\mathbf{x}_{ij}(\xi_\nu), t), \mathbf{n}_{ij} \right) \mathbf{L}(\psi_i(\mathbf{x}_{ij}(\xi_\nu))).$$

For the nodal version (1.83) of the DG scheme on triangular elements we then have the following SBP property.

**Theorem 1.15.** *For the DG scheme (1.83) we have the subsequent assertions which yield a specific form of an SBP property on a triangular grid.*

*First, the mass matrix  $\mathbf{M}$  is invertible. Let the matrix  $\tilde{\mathbf{L}} \in \mathbb{R}^{N_I \times n_{inner}}$  be defined by encoding the evaluation of the Lagrange basis functions at the quadrature nodes, i.e.  $\tilde{L}_{k\mu} = L_k \circ \psi_i^{-1}(r_\mu, s_\mu)$  and let  $\mathbf{P} = \mathbf{M}^{-1} \tilde{\mathbf{L}} \mathbf{W}$ .*

1. *The matrix  $\tilde{\mathbf{P}} = \tilde{\mathbf{L}}^T \mathbf{P}$  defines a projection in  $\mathbb{R}^{n_{inner}}$ .*

*In addition, we have the identity matrix  $\mathbf{P} \tilde{\mathbf{L}}^T = \mathbf{I} \in \mathbb{R}^{N_I \times N_I}$ .*

2. *The discrete derivative operators  $\tilde{\mathbf{D}}_x, \tilde{\mathbf{D}}_y$  with respect to the quadrature nodes may be rewritten in terms of the DG nodal set as*

$$\tilde{\mathbf{D}}_x^T \mathbf{W} = \mathbf{D}_x^T \mathbf{M} \mathbf{P} \quad \text{and} \quad \tilde{\mathbf{D}}_y^T \mathbf{W} = \mathbf{D}_y^T \mathbf{M} \mathbf{P}, \quad (1.85)$$

*where*

$$D_{x,jk} = \frac{\partial}{\partial x} L_k(\mathbf{x}_j) \quad \text{and} \quad D_{y,jk} = \frac{\partial}{\partial y} L_k(\mathbf{x}_j). \quad (1.86)$$

3. *The operators  $\mathbf{D}_x, \mathbf{D}_y$  are first-derivative SBP operators of degree  $N$  fulfilling the accuracy conditions*

$$\mathbf{D}_x \mathbf{p}_{\alpha,\beta} = \alpha \mathbf{p}_{\alpha-1,\beta}, \quad \mathbf{D}_y \mathbf{p}_{\alpha,\beta} = \beta \mathbf{p}_{\alpha,\beta-1}, \quad \alpha + \beta \leq N, \quad (1.87)$$

*for polynomial functions  $p_{\alpha,\beta}(x, y) = x^\alpha y^\beta$  and their corresponding nodal values  $\mathbf{p}_{\alpha,\beta}$ , as well as the SBP properties*

$$\mathbf{M} \mathbf{D}_x + \mathbf{D}_x^T \mathbf{M} = \mathbf{B}_x, \quad \mathbf{M} \mathbf{D}_y + \mathbf{D}_y^T \mathbf{M} = \mathbf{B}_y, \quad (1.88)$$

*with  $B_{x,km} = \langle L_k L_m n_x \rangle_{\partial\tau_i} = \int_{\partial\tau_i} L_k L_m n_x d\sigma$  and  $B_{y,km} = \langle L_k L_m n_y \rangle_{\partial\tau_i} = \int_{\partial\tau_i} L_k L_m n_y d\sigma$ .*

*Proof.* Due to the exactness of the quadrature rule encoded in  $\mathbf{W}$  for polynomials in  $P^N(\tau_i)$ , we obviously have  $\mathbf{M} = \tilde{\mathbf{L}} \mathbf{W} \tilde{\mathbf{L}}^T$ . In particular,  $\mathbf{M}$  is symmetric, positive definite and thus invertible. In addition, the itemized assertions are proven as follows.

1. Defining  $\mathbf{P} = \mathbf{M}^{-1}\tilde{\mathbf{L}}\mathbf{W}$  and  $\tilde{\mathbf{P}} = \tilde{\mathbf{L}}^T\mathbf{P}$ , we have

$$\tilde{\mathbf{P}}^2 = (\tilde{\mathbf{L}}^T\mathbf{P})^2 = \tilde{\mathbf{L}}^T\mathbf{M}^{-1}\underbrace{\tilde{\mathbf{L}}\mathbf{W}\tilde{\mathbf{L}}^T}_{=\mathbf{M}}\mathbf{M}^{-1}\tilde{\mathbf{L}}\mathbf{W} = \tilde{\mathbf{L}}^T\mathbf{M}^{-1}\tilde{\mathbf{L}}\mathbf{W} = \tilde{\mathbf{P}}.$$

Furthermore, we have  $\mathbf{P}\tilde{\mathbf{L}}^T = \mathbf{M}^{-1}\tilde{\mathbf{L}}\mathbf{W}\tilde{\mathbf{L}}^T = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$ .

2. Due to the expansion  $\frac{\partial}{\partial x}L_k(\mathbf{x}) = \sum_j \frac{\partial}{\partial x}L_k(\mathbf{x}_j)L_j(\mathbf{x})$  we have

$$\tilde{D}_{x,\mu k} = \frac{\partial}{\partial x}L_k(\psi_i^{-1}(r_\mu, s_\mu)) = \sum_j \frac{\partial}{\partial x}L_k(\mathbf{x}_j)L_j(\psi_i^{-1}(r_\mu, s_\mu)) = \sum_j D_{x,jk}\tilde{L}_{j\mu}$$

and hence  $\tilde{\mathbf{D}}_x = \tilde{\mathbf{L}}^T\mathbf{D}_x$ . An analogous derivation for the variable  $y$  yields  $\tilde{\mathbf{D}}_y = \tilde{\mathbf{L}}^T\mathbf{D}_y$ . Using the definition of the matrix  $\mathbf{P} = \mathbf{M}^{-1}\tilde{\mathbf{L}}\mathbf{W}$  this yields

$$\tilde{\mathbf{D}}_x^T\mathbf{W} = \mathbf{D}_x^T\tilde{\mathbf{L}}\mathbf{W} = \mathbf{D}_x^T\mathbf{M}\mathbf{P} \quad \text{and} \quad \tilde{\mathbf{D}}_y^T\mathbf{W} = \mathbf{D}_y^T\tilde{\mathbf{L}}\mathbf{W} = \mathbf{D}_y^T\mathbf{M}\mathbf{P}.$$

3. Regarding the accuracy conditions (1.87), derivation of the multivariate polynomial function  $p_{\alpha,\beta}(x, y) = x^\alpha y^\beta$  with respect to  $x$  yields  $\frac{\partial}{\partial x}p_{\alpha,\beta}(x, y) = \alpha p_{\alpha-1,\beta}(x, y)$ . Furthermore, since  $p_{\alpha,\beta} \in P^N(\tau_i)$ , it may be exactly represented using Lagrange interpolation based on the nodal set  $\{\mathbf{x}_j\}_{j=1,\dots,N_I}$ , i.e.

$$p_{\alpha,\beta}(x, y) = \sum_{j=1}^{N_I} x_j^\alpha y_j^\beta \frac{\partial}{\partial x} L_j(x, y).$$

Derivation with respect to  $x$  and evaluation at a specific node  $\mathbf{x}_k$  then yields

$$\alpha p_{\alpha-1,\beta}(x_k, y_k) = \frac{\partial}{\partial x}p_{\alpha,\beta}(x_k, y_k) = \sum_{j=1}^{N_I} x_j^\alpha y_j^\beta \frac{\partial}{\partial x} L_j(x_k, y_k),$$

thus by definition of  $\mathbf{D}_x$  via  $D_{x,kj} = \frac{\partial}{\partial x}L_j(x_k, y_k)$ , this is represented by the vector valued equation  $\alpha \mathbf{p}_{\alpha-1,\beta} = \mathbf{D}_x \mathbf{p}_{\alpha,\beta}$ . An analogous derivation with respect to the variable  $y$  yields the respective accuracy condition  $\beta \mathbf{p}_{\alpha,\beta-1} = \mathbf{D}_y \mathbf{p}_{\alpha,\beta}$ .

In order to prove the SBP properties (1.88), we first show

$$\mathbf{M}\mathbf{D}_x = \mathbf{Q}_x \quad \text{and} \quad \mathbf{M}\mathbf{D}_y = \mathbf{Q}_y,$$

with  $\mathbf{Q}_x, \mathbf{Q}_y \in \mathbb{R}^{N_I \times N_I}$  given by their respective entries  $Q_{x,km} = \int_{\tau_i} L_k(\mathbf{x}) \frac{\partial}{\partial x} L_m(\mathbf{x}) d\mathbf{x}$  and  $Q_{y,km} = \int_{\tau_i} L_k(\mathbf{x}) \frac{\partial}{\partial y} L_m(\mathbf{x}) d\mathbf{x}$ . This is proven by exactness of the quadrature rule encoded in  $\mathbf{W}$  for the polynomials  $L_k \frac{\partial}{\partial x} L_m$  where  $k, m = 1, \dots, N_I$ , since we have  $\mathbf{Q}_x = \tilde{\mathbf{L}}\mathbf{W}\tilde{\mathbf{D}}_x$  and hence  $\mathbf{Q}_x = \tilde{\mathbf{L}}\mathbf{W}\tilde{\mathbf{L}}^T\mathbf{D}_x = \mathbf{M}\mathbf{D}_x$ . The same procedure for  $\mathbf{Q}_y$  yields  $\mathbf{Q}_y = \mathbf{M}\mathbf{D}_y$ .

Furthermore, partial integration yields  $\mathbf{Q}_x + \mathbf{Q}_x^T = \mathbf{B}_x$ , where  $\mathbf{B}_x$  is given by the entries

$$B_{x,km} = \int_{\tau_i} \left( L_k(\mathbf{x}) \frac{\partial}{\partial x} L_m(\mathbf{x}) d\mathbf{x} + L_m(\mathbf{x}) \frac{\partial}{\partial x} L_k(\mathbf{x}) d\mathbf{x} \right) d\mathbf{x} = \int_{\partial\tau_i} L_k L_m n_x d\sigma.$$

In addition, the degree of exactness of the boundary quadrature rule yields  $B_{x,km} = \langle L_k L_m n_x \rangle_{\partial\tau_i}$ . An analogous derivation with respect to the variable  $y$  provides the corresponding assertion  $\mathbf{M}\mathbf{D}_y + \mathbf{D}_y^T\mathbf{M} = \mathbf{B}_y$ .

□

In summary, substituting (1.85) and (1.88) in (1.83), we obtain

$$\mathbf{M} \frac{d}{dt} \mathbf{u} + \mathbf{MD}_x \mathbf{P} \tilde{\mathbf{f}}_1 + \mathbf{MD}_y \mathbf{P} \tilde{\mathbf{f}}_2 = \mathbf{B}_x \mathbf{P} \tilde{\mathbf{f}}_1 + \mathbf{B}_y \mathbf{P} \tilde{\mathbf{f}}_2 - \langle f^*(u_h^-, u_h^+, \mathbf{n}) \mathbf{L} \rangle_{\partial\tau_i}$$

In fact, in the above formulation, the matrix  $\mathbf{P}$  defined in Theorem 1.15 is used to project the flux values at the quadrature nodes to the DG nodal set to obtain the nodal values  $\mathbf{P} \tilde{\mathbf{f}}_j$ , for  $j = 1, 2$ . Based on these nodal values, we may define the vector  $\mathbf{f}_h = (f_{h,1}, f_{h,2})^T \in (P^N(\tau_i))^2$ , where the polynomial functions  $f_{h,j}$ ,  $j = 1, 2$ , result from interpolation of the nodal values  $\mathbf{P} \tilde{\mathbf{f}}_j$ , using the DG basis functions. Thus, by the definition of the matrices  $\mathbf{B}_x, \mathbf{B}_y$ , and using short notation for numerical integration over the element boundary, we have

$$\mathbf{B}_x \mathbf{P} \tilde{\mathbf{f}}_1 + \mathbf{B}_y \mathbf{P} \tilde{\mathbf{f}}_2 = \langle (\mathbf{f}_h \cdot \mathbf{n}) \mathbf{L} \rangle_{\tau_i}.$$

Therefore, the DG scheme on triangular grids in SBP form is given by

$$\frac{d}{dt} \mathbf{u} + \mathbf{D}_x \mathbf{P} \tilde{\mathbf{f}}_1 + \mathbf{D}_y \mathbf{P} \tilde{\mathbf{f}}_2 = \mathbf{M}^{-1} \langle (\mathbf{f}_h \cdot \mathbf{n} - f^*(u_h^-, u_h^+, \mathbf{n})) \mathbf{L} \rangle_{\partial\tau_i}. \quad (1.89)$$

**Remark 1.16.** *Regarding the derivative operators  $\mathbf{D}_x, \mathbf{D}_y$ , the specific form of the SBP property given in (1.88) directly corresponds to the definition of multidimensional SBP operators on simplex elements given by Hicken et al. in [77]. However, for the DG scheme on triangular grids using a warped tensor-product quadrature rule, the above form of the SBP scheme (1.89) slightly differs from [77] as only one set of nodes is considered in that work. Here, the set of quadrature nodes represents an additional nodal set and the matrix  $\mathbf{P}$  is used to transfer the corresponding function values to the nodal set used for the actions of  $\mathbf{D}_x, \mathbf{D}_y$ .*

**Remark 1.17.** *A pre-integrated nodal DG scheme on triangular grids is obtained by evaluating the flux function at the interpolation nodes, i.e. replacing  $\tilde{\mathbf{f}}_l$  in (1.89) by*

$$\mathbf{f}_l = (f_l(u_h(\mathbf{x}_1, t)), \dots, f_l(u_h(\mathbf{x}_{N_I}, t)))^T, \quad l = 1, 2,$$

*considering  $\mathbf{f}_h$  as the flux polynomial obtained by interpolation, and setting the projection matrix to the identity matrix, i.e.  $\mathbf{P} = \mathbf{I} \in \mathbb{R}^{N_I \times N_I}$ . For the resulting scheme given by*

$$\frac{d}{dt} \mathbf{u} + \mathbf{D}_x \mathbf{f}_1 + \mathbf{D}_y \mathbf{f}_2 = \mathbf{M}^{-1} \langle (\mathbf{f}_h \cdot \mathbf{n} - f^*(u_h^-, u_h^+, \mathbf{n})) \mathbf{L} \rangle_{\partial\tau_i}, \quad (1.90)$$

*the third assertion of Theorem 1.15 regarding the SBP property of the nodal DG scheme obviously still holds. The pre-integrated triangular grid nodal DG scheme (1.90) now directly falls into the class of SBP schemes on triangular grids considered in [77].*

### 1.3 Energy stability of flux reconstruction schemes

Closely linked to the theory of SBP schemes is the development of energy stable schemes of flux reconstruction type. The flux reconstruction (FR) approach, later denoted as correction

procedure via reconstruction (CPR) in [87] was initially introduced by Huynh in [85] to generalize various high order schemes used in computational fluid dynamics via their differential formulation. The FR approach unifies the DG method and several other popular schemes as shown in [85, 206], such as the original staggered-grid scheme by Kopriva and Kolias [102], the spectral difference (SD) scheme by Liu et al., see e.g. [118] and the spectral volume (SV) method [207].

### Flux reconstruction (FR) schemes for conservation laws in 1D

Due to the similarity of the flux reconstruction approach to nodal DG schemes, some of the notation introduced in Section 1.2.1 will be reused in the following. Again, the computational domain is divided into non-overlapping cells which can be mapped to a reference element on which a set of nodes is chosen. For the approximation of a one-dimensional conservation law

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(x, t) = 0$$

as in (1.40), the flux reconstruction approach constructs an approximate solution  $u_h$  which is a piecewise polynomial function. On each cell  $I_i$ , the polynomial function  $u_h^i(\xi, t) = u_h(\Lambda_i(\xi), t)$ , obtained by a transformation to the reference cell  $[-1, 1]$  using the map  $\Lambda_i$  is then required to satisfy the equation

$$\frac{\partial u_h^i}{\partial t}(\xi, t) = -\frac{2}{\Delta x} \left[ \frac{\partial f_h^i}{\partial \xi}(\xi, t) + \underbrace{(f_i^* - f_h^i(-1, t))}_{f_{CL}} g_L'(\xi) + \underbrace{(f_{i+1}^* - f_h^i(1, t))}_{f_{CR}} g_R'(\xi) \right]. \quad (1.91)$$

Herein, the function  $f_h^i$  is a polynomial of degree  $N$  obtained via interpolation of the nodal values  $f_\nu = f(u_h^i(\xi_\nu, t))$  with  $\xi_\nu, \nu = 1, \dots, N+1$  and the quantities  $f_{CL}$  and  $f_{CR}$  denote the flux jumps at the left and right interfaces containing the values of a numerical flux function  $f^*$ . Furthermore,  $g_L$  and  $g_R$  are so-called left and right correction polynomials of degree  $N+1$  which are required to fulfill the conditions

$$g_L(-1) = g_R(1) = 1 \quad \text{and} \quad g_R(-1) = g_L(1) = 0. \quad (1.92)$$

Usually, symmetry of the correction polynomials is enforced by demanding  $g_R(\xi) = g_L(-\xi)$ .

Regarding the specific choice of correction polynomials, it has been shown in [85] that the classical 1D nodal DG scheme without mass lumping is recovered if  $g_L$  and  $g_R$  are the right and left Radau polynomials, respectively. For the left correction polynomial, this means that  $g_L$  vanishes at the  $N+1$  Legendre-Gauss-Radau quadrature nodes including the right end point. Together with the condition  $g_L(-1) = 1$  and the symmetry property  $g_R(\xi) = g_L(-\xi)$ , the left and right correction polynomials of degree  $N+1$  are thereby uniquely defined.

Furthermore, Huynh [85] showed that SD type schemes can be recovered for linear flux functions if  $g_L$  and  $g_R$  have coincident zeros in the interior of the reference cell. Due to the required symmetry of the left and right correction functions, this is the case if the zeros are located symmetrically in  $[-1, 1]$  with respect to the origin. In fact, the original idea of the SD scheme is to represent the numerical solution on each cell as a polynomial of degree  $N$  and the

corresponding flux as a polynomial of degree  $N + 1$ , obtained by interpolation on a set of flux collocation points, also called flux points. The flux points are required to contain a sufficient number of boundary nodes which incorporate the values of a chosen single-valued numerical flux function. Hence, in the one-dimensional case, the cell boundaries and  $N$  interior points are needed to represent the flux polynomial on each cell and the global representation of the flux is continuous by construction. The cell-wise derivative  $\frac{\partial u_h^i}{\partial t}$  is then obtained directly by differentiating the flux polynomial with respect to the space coordinate. Considering the representation as a flux reconstruction scheme based on the formulation (1.91), the interior flux points of the SD scheme are thus given as the coincident zeros of  $g_L$  and  $g_R$ . Therefore, the SD schemes within the FR framework are based on a very economical formulation of the reconstructed flux given by  $f_h^i + f_{CL}g_L + f_{CR}g_R$  since the effect of the correction terms may be reduced to the boundary nodes. Unfortunately, it has henceforth been difficult to construct SD schemes devoid of weak instabilities, in particular on triangular grids. In this regard, van Abeele et al. [196] showed that the stability of an SD scheme generally depends only on the choice of flux collocation points and that in one space dimension, the often used Chebyshev-Gauss-Lobatto points result in unstable SD schemes for orders of accuracy higher than two. Furthermore, they were unable to construct stable SD schemes of higher than second order on triangular grids.

On the other hand, in [91], Jameson proved the stability of one-dimensional SD schemes of any order of accuracy for the linear advection equation if the zeros of the corresponding Legendre polynomial are chosen as the interior flux points. An extension of this analysis lead to the development of energy stable FR (ESFR) schemes. This particular class of flux reconstruction schemes was identified by Vincent, Castonguay and Jameson in [202], and in reference to Huynh [85], its members are also referred to as VCJH schemes. In one space dimension, the class of ESFR schemes contains precisely one SD type scheme which is also the only SD type scheme that Huynh found to be stable by using von Neumann analysis in [85].

A close connection between ESFR and SBP schemes can be found by rewriting the flux reconstruction form (1.91) in its matrix-vector representation which was used by Jameson and Allaneau in [3] in order to interpret ESFR schemes as filtered DG methods. The matrix-vector formulation is established by expanding the numerical solution  $u_h^i$ , the flux polynomial  $f_h^i$ , and the derivatives  $g'_L, g'_R$  of the correction polynomials in the same basis which is given by the Lagrange polynomials  $L_k$  corresponding to the given nodes  $\xi_k$ ,  $k = 1, \dots, N + 1$ . Denoting by  $\mathbf{u}, \mathbf{f}, \mathbf{g}'_L, \mathbf{g}'_R$  the respective nodal representations on the cell  $I_i$  and using the derivative operator  $\mathbf{D}$  defined by the entries  $D_{jk} = L'_k(\xi_j)$ , analogously to (1.52), we have

$$\frac{\Delta x}{2} \frac{d\mathbf{u}}{dt} + \mathbf{D}\mathbf{f} = -f_{CL}\mathbf{g}'_L - f_{CR}\mathbf{g}'_R. \quad (1.93)$$

Multiplying (1.93) from the left by the mass matrix  $\mathbf{M}$  with  $M_{jk} = \int_{-1}^1 L_j L_k d\xi$  as in (1.47), we obtain

$$\frac{\Delta x}{2} \mathbf{M} \frac{d\mathbf{u}}{dt} + \mathbf{S}\mathbf{f} = -f_{CL}\mathbf{M}\mathbf{g}'_L - f_{CR}\mathbf{M}\mathbf{g}'_R =: RHS_{FR}, \quad (1.94)$$

with  $\mathbf{S} = \mathbf{MD}$  matching the stiffness matrix defined in (1.48) and  $RHS_{FR}$  denoting the right-hand side of the flux reconstruction formulation (1.94).

Temporarily neglecting the indices  $L$  and  $R$  of the correction polynomials  $g_L$  and  $g_R$ , the vector  $\mathbf{g}'$  is the nodal representation of the polynomial  $g'$  using the Lagrange basis  $L_k$  which can be collected into the vector valued function  $\mathbf{L} = (L_1, \dots, L_{N+1})^T$  as in (1.49). Therefore, multiplication by  $\mathbf{M}$  results in

$$\mathbf{M} \mathbf{g}' = \int_{-1}^1 g' \mathbf{L} d\xi = [g \mathbf{L}]_{-1}^1 - \int_{-1}^1 g \mathbf{L}' d\xi$$

and due to the conditions (1.92) on the boundary values of  $g_L$  and  $g_R$ , we have the decomposition of the FR right-hand side

$$RHS_{FR} = \underbrace{f_{CL}\mathbf{L}(-1) - f_{CR}\mathbf{L}(1)}_{RHS_{DG}} + \underbrace{\int_{-1}^1 (f_{CL}g_L + f_{CR}g_R)\mathbf{L}' d\xi}_{\text{Deviation from DG}} \quad (1.95)$$

into the right-hand side of the DG scheme  $RHS_{DG}$  and the deviation from DG.

In [3], the derivation of energy stable FR schemes is based on the above equivalent matrix-vector formulations (1.93) and (1.94). Given a symmetric positive semi-definite matrix  $\mathbf{K}$  with  $\mathbf{K}\mathbf{D} = \mathbf{0}$ , left multiplication of (1.93) by  $\mathbf{K}$  and subsequent addition to (1.94) with  $RHS_{FR}$  given by (1.95) yields

$$\begin{aligned} \frac{\Delta x}{2} (\mathbf{M} + \mathbf{K}) \frac{d\mathbf{u}}{dt} + \mathbf{S}\mathbf{f} &= RHS_{DG} \\ &+ f_{CL} \left( \int_{-1}^1 g_L \mathbf{L}' d\xi - \mathbf{K} \mathbf{g}'_L \right) + f_{CR} \left( \int_{-1}^1 g_R \mathbf{L}' d\xi - \mathbf{K} \mathbf{g}'_R \right). \end{aligned} \quad (1.96)$$

Regarding the ESFR schemes constructed by Vincent et al. in [202], the correction polynomials  $g_L$  and  $g_R$  are chosen such that all terms on the right-hand side of (1.96) other than  $RHS_{DG}$  vanish. Therefore, the ESFR scheme corresponds to a substitution of the DG mass matrix  $\mathbf{M}$  within a classical DG scheme by the symmetric positive definite matrix  $\mathbf{M} + \mathbf{K}$ . This leads to a filtered DG scheme as uncovered by Allaneau and Jameson in [3].

In addition, noting the conformity to SBP schemes is of particular worth. In fact, since an ESFR scheme has the representation

$$\frac{\Delta x}{2} (\mathbf{M} + \mathbf{K}) \frac{d\mathbf{u}}{dt} + \mathbf{S}\mathbf{f} = RHS_{DG}, \quad (1.97)$$

with the derivative operator  $\mathbf{D}$  corresponding to the DG scheme and a suitable symmetric positive semi-definite matrix  $\mathbf{K}$  with  $\mathbf{K}\mathbf{D} = \mathbf{0} = \mathbf{D}^T\mathbf{K}$ , the ESFR scheme inherits its SBP property directly from the DG scheme by considering the equality

$$(\mathbf{M} + \mathbf{K})\mathbf{D} + \mathbf{D}^T(\mathbf{M} + \mathbf{K}) = \mathbf{MD} + \mathbf{D}^T\mathbf{M}.$$

Thus, Theorem 1.5 carries over to one-dimensional ESFR schemes.

### ESFR schemes on triangular grids

In the following, we will show that the ESFR schemes on triangular grids which have been identified by Castonguay et al. in [36] satisfy an SBP property as well. For this purpose, we

use a reformulation of ESFR schemes on triangular grids in matrix-vector form analogous to the one given by Allaneau and Jameson [3] for the one-dimensional case. This again allows us to represent triangular-grid ESFR schemes as filtered nodal DG schemes which additionally fulfill an SBP property of the type given in equation (1.88).

Reusing the notation introduced for nodal DG schemes on triangular grids, an FR scheme approximating the two-dimensional scalar hyperbolic conservation law (1.71) on a triangular element has the form

$$\frac{d}{dt} u_h + \nabla \cdot \mathbf{f}_h = - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \nabla \cdot \mathbf{h}_{j\nu}, \quad (1.98)$$

where we neglected the index  $i$  of the triangular element  $\tau_i$  and where the flux polynomial  $\mathbf{f}_h \in (P^N(\tau_i))^2$  is obtained by interpolating  $\mathbf{f}(u_h(\mathbf{x}, t))$  on a suitable set of interpolation points on  $\tau_i$ . When using a polynomial space of degree  $N$ , the number of interpolation points on a triangular element is  $N_I = \frac{1}{2}(N+1)(N+2)$ , analogously to nodal DG schemes. The right-hand side of (1.98) is a correction term containing the flux jumps  $f_{C,j\nu}$  and correction polynomials  $\mathbf{h}_{j\nu}$  corresponding to a set of pre-determined flux points on the element boundary  $\partial\tau_i$ . For an FR scheme of degree  $N$ , the flux points at each triangle edge are chosen as the  $N+1$  Legendre-Gauss integration points. Hence, the notation for triangular grid nodal DG schemes introduced in Section 1.2.5 may be used to write the correction term in a more precise form. Hereby, the correction polynomial  $\mathbf{h}_{j\nu}$  corresponds to the node  $\mathbf{x}_{ij}(\xi_\nu)$  on the edge  $\Gamma_{ij}$  of  $\tau_i$ . Furthermore, similar to the FR schemes in 1D, the flux jumps at edge nodes are given by

$$f_{C,j\nu} = f^* \left( u_h^i(\mathbf{x}_{ij}(\xi_\nu), t), u_h^{ij}(\mathbf{x}_{ij}(\xi_\nu), t), \mathbf{n}_{ij} \right) - \mathbf{f}_{h,j\nu} \cdot \mathbf{n}_{ij}, \quad (1.99)$$

where  $\mathbf{f}_{h,j\nu} = \mathbf{f}_h(\mathbf{x}_{ij}(\xi_\nu))$ .

Now, the gist of constructing FR schemes is to precisely specify the correction polynomials  $\mathbf{h}_{j\nu}$  for each node on the element boundary  $\tau_i$ . Hereby, the correction polynomials of ESFR schemes are required to fulfill the properties

$$\begin{aligned} \nabla \cdot \mathbf{h}_{j\nu} &\in P^N \tau_i, \\ \mathbf{h}_{j\nu} \cdot \mathbf{n}_{ik} &\in P^N \Gamma_{ik}, \end{aligned} \quad (1.100)$$

making each correction polynomial  $\mathbf{h}_{j\nu}$  a member of the Raviart-Thomas space  $RT_N(\tau_i)$  of order  $N$ . This space is the smallest polynomial space such that the divergence maps  $RT_N(\tau_i)$  onto  $P^N(\tau_i)$ . Furthermore, the correction polynomials are supposed to satisfy

$$\mathbf{h}_{j\nu}(x_{ik}(\xi_\mu)) \cdot \mathbf{n}_{ik} = \begin{cases} 1 & \text{if } k = j \text{ and } \mu = \nu, \\ 0 & \text{else.} \end{cases} \quad (1.101)$$

In order to transfer the Allaneau & Jameson procedure deriving ESFR schemes to the triangular grid case, the FR scheme (1.98) is first rewritten in a matrix-vector formulation.

For this purpose, let  $\mathbf{u}$  denote the vector of nodal values of the approximate solution and  $\mathbf{f}$  the vector of flux values at the same set of interpolation points. Furthermore, as before, the matrices  $\mathbf{D}_x$  and  $\mathbf{D}_y$  are built by evaluating the spatial derivatives of the Lagrange polynomials

corresponding to the interpolation points, i.e.  $\mathbf{D}_x$  and  $\mathbf{D}_y$  are defined as in (1.86). Since by construction, the divergence  $\Phi_{j\nu} = \nabla \cdot \mathbf{h}_{j\nu}$  of any correction function  $\mathbf{h}_{j\nu}$  is a polynomial in  $P^N(\tau_i)$ , we may represent  $\Phi_{j\nu}$  by the vector  $\Phi_{j\nu}$  of its nodal values at the given interpolation points. In matrix-vector form, the FR scheme (1.98) is therefore given by

$$\frac{d}{dt} \mathbf{u} + \mathbf{D}_x \mathbf{f}_1 + \mathbf{D}_y \mathbf{f}_2 = - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \Phi_{j\nu}.$$

Next, we multiply the above equation from the left by the mass matrix  $\mathbf{M}$  corresponding to the Lagrange basis on the triangular element, i.e.  $\mathbf{M}$  is defined as in (1.84). This yields

$$\mathbf{M} \frac{d}{dt} \mathbf{u} + \mathbf{M} \mathbf{D}_x \mathbf{f}_1 + \mathbf{M} \mathbf{D}_y \mathbf{f}_2 = - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \mathbf{M} \Phi_{j\nu}. \quad (1.102)$$

Furthermore, we have

$$\mathbf{M} \Phi_{j\nu} = \int_{\tau_i} \Phi_{j\nu} \mathbf{L} d\mathbf{x} = \int_{\partial\tau_i} (\mathbf{h}_{j\nu} \cdot \mathbf{n}) \mathbf{L} d\sigma - \int_{\tau_i} \mathbf{h}_{j\nu} \cdot \nabla \mathbf{L} d\mathbf{x}, \quad (1.103)$$

where  $\mathbf{h}_{j\nu} \cdot \nabla \mathbf{L} = (\mathbf{h}_{j\nu} \cdot \nabla L_1, \dots, \mathbf{h}_{j\nu} \cdot \nabla L_{N_I})^T$ .

Analogously to the 1D case, for the identification of ESFR schemes, a matrix  $\mathbf{K}$  is constructed such that  $\mathbf{K} \mathbf{D}_x = \mathbf{0} = \mathbf{K} \mathbf{D}_y$ . On left multiplication of (1.98) with  $\mathbf{K}$ , we then have

$$\mathbf{K} \frac{d}{dt} \mathbf{u} = - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \mathbf{K} \Phi_{j\nu}. \quad (1.104)$$

Adding up equations (1.102) and (1.104) results in

$$(\mathbf{M} + \mathbf{K}) \frac{d}{dt} \mathbf{u} + \mathbf{M} \mathbf{D}_x \mathbf{f}_1 + \mathbf{M} \mathbf{D}_y \mathbf{f}_2 = - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} (\mathbf{M} + \mathbf{K}) \Phi_{j\nu}, \quad (1.105)$$

while taking into account (1.103) yields

$$\sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \mathbf{M} \Phi_{j\nu} = \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \int_{\partial\tau_i} (\mathbf{h}_{j\nu} \cdot \mathbf{n}) \mathbf{L} d\sigma - \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \int_{\tau_i} \mathbf{h}_{j\nu} \cdot \nabla \mathbf{L} d\mathbf{x}.$$

Since the  $N+1$  flux points on each triangle edge are precisely the Legendre-Gauss integration points transferred to the respective edge, the corresponding quadrature rule exactly integrates polynomials of degree  $2N+1$ . This is sufficiently accurate in order to evaluate the first term on the right-hand side of the above equation. Using the conditions (1.101) on the pointwise values of the correction polynomials, we thus have

$$- \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \int_{\partial\tau_i} (\mathbf{h}_{j\nu} \cdot \mathbf{n}) \mathbf{L} d\sigma = - \sum_{j=1}^3 \frac{|\Gamma_{ij}|}{2} \sum_{\nu=1}^{n_{edge}} \omega_\nu f_{C,j\nu} \mathbf{L}(x_{ij}(\xi_\nu)).$$

Using the definition of the flux jumps given in (1.99) shows that this is precisely the right-hand side of a pre-integrated nodal DG scheme on triangular grids obtained by multiplying (1.90) from left by  $\mathbf{M}$ , i.e.

$$RHS_{DG} = - \sum_{j=1}^3 \frac{|\Gamma_{ij}|}{2} \sum_{\nu=1}^{n_{edge}} \omega_\nu f_{C,j\nu} \mathbf{L}(x_{ij}(\xi_\nu)) = \langle (\mathbf{f}_h \cdot \mathbf{n} - f^*(u_h^-, u_h^+, \mathbf{n})) \mathbf{L} \rangle_{\partial\tau_i}.$$

With the above considerations, we may express the FR scheme (1.105) by

$$\begin{aligned} (\mathbf{M} + \mathbf{K}) \frac{d}{dt} \mathbf{u} + \mathbf{MD}_x \mathbf{f}_1 + \mathbf{MD}_y \mathbf{f}_2 &= RHS_{DG} \\ &+ \sum_{j=1}^3 \sum_{\nu=1}^{n_{edge}} f_{C,j\nu} \left( \int_{\tau_i} \mathbf{h}_{j\nu} \cdot \nabla \mathbf{L} d\mathbf{x} - \mathbf{K} \Phi_{j\nu} \right). \end{aligned} \quad (1.106)$$

Analogously to the one-dimensional case, the triangular grid ESFR schemes presented by Castonguay et al. in [36] are obtained by constructing correction polynomials such that the last term on the right-hand side of (1.106) vanishes, which means that the conditions

$$\int_{\tau_i} \mathbf{h}_{j\nu} \cdot \nabla \mathbf{L} d\mathbf{x} - \mathbf{K} \Phi_{j\nu}, \quad j = 1, 2, 3, \quad \nu = 1, \dots, N+1,$$

are to be fulfilled. With  $RHS_{DG}$  the only remaining term on the right-hand side of (1.106), an ESFR scheme obviously constitutes a filtered DG scheme analogously to the one-dimensional case given in (1.97). Furthermore, since by the specific construction of  $\mathbf{K}$  we have

$$(\mathbf{M} + \mathbf{K}) \mathbf{D}_x + \mathbf{D}_x^T (\mathbf{M} + \mathbf{K}) = \mathbf{MD}_x + \mathbf{D}_x^T \mathbf{M} \quad \text{and} \quad (\mathbf{M} + \mathbf{K}) \mathbf{D}_y + \mathbf{D}_y^T (\mathbf{M} + \mathbf{K}) = \mathbf{MD}_y + \mathbf{D}_y^T \mathbf{M},$$

with the matrices  $\mathbf{M}, \mathbf{D}_x, \mathbf{D}_y$  matching the DG case considered in Section 1.2.5, the ESFR scheme therefore fulfills an SBP property analogous to the SBP property (1.88) of a triangular grid nodal DG scheme.

## 1.4 Kinetic energy preserving DG schemes for the Euler- and Navier-Stokes equations

In the context of numerical methods for conservation laws, the preservation of the primary conserved quantities usually is a minimum requirement. In addition, the balance of secondary quantities may be desirable as well, such as kinetic energy in case of the Euler equations of gas dynamics. This also extends to the simulation of viscous flow. In fact, particularly for the simulation of turbulent flows, an accurate simulation of the kinetic energy is generally desired, see e.g. [137, 200, 186, 136, 63, 93]. For finite volume methods, specifically designed numerical fluxes as in [92] may guarantee either preservation of entropy or energy. Similar construction principles leading to mimetic schemes which guarantee enhanced conservation properties can be found also in the context of shallow water flows. For example, energy conservation is desired for simulations involving rapidly varied flow e.g. due to large gradients in bathymetry [181], or total energy is conserved in addition to mass and momentum as in [79].

Another benefit of the preservation of secondary quantities is given by additional energy estimates which may be obtained, potentially enabling stability of the scheme without or with a reduced amount of artificial dissipation. This in turn can improve the accuracy of both viscous and inviscid flow computations which are otherwise often compromised by the dissipative mechanisms used for shock capturing. Of course, this feature is particularly desirable for higher order methods and is reflected by SBP finite difference schemes applied to skew-symmetric forms of conservation laws. While the skew-symmetric form is used to enforce enhanced conservation of specific secondary quantities, the SBP property guarantees conservation of the primary conserved quantities.

Closely linked to the framework of SBP finite difference schemes, Gassner [61] constructed a kinetic energy preserving discontinuous Galerkin scheme in one space dimension using Legendre-Gauss-Lobatto nodes based on a skew-symmetric formulation of the Euler equations. This construction rests upon the classical SBP property of the Legendre-Gauss-Lobatto DG scheme. However, as discussed in Section 1.2.1, the generalized SBP property is not restricted to Legendre-Gauss-Lobatto nodes. Therefore, in [150], a kinetic energy preserving DG scheme using Legendre-Gauss nodes has been constructed which is similar in spirit to the skew-symmetric DG scheme [61] but builds upon the generalized SBP property shown in Theorem 1.5.

In Section 1.4.1, the construction of the 1D KEP-DG scheme on Legendre-Gauss nodes as in [150] is described in detail. Hereby, the required form of the corresponding boundary correction terms in the skew-symmetric formulation leading to a conservative and consistent scheme is theoretically investigated. In fact, for a Legendre-Gauss point distribution, boundary terms require special attention. Whereas the DG scheme on Legendre-Gauss nodes yields a diagonal mass matrix and hence a diagonal norm SBP operator precisely as for Legendre-Gauss-Lobatto nodes, the interface operator is not diagonal since Legendre-Gauss nodes do not include the interval boundaries of a grid cell. Hence, the skew-symmetric DG scheme constructed in [61] can not directly be combined with Legendre-Gauss nodes. In this regard, stability issues will be pointed out which may arise when using a combination of skew-symmetric terms with inconsistent boundary treatment that disagrees with exclusively interior nodal sets.

In numerical experiments we study the order of convergence for smooth solutions, the kinetic energy balance and the behavior of different variants of the scheme applied to an acoustic pressure wave and a viscous shock tube. Since quadrature rules based on the Legendre-Gauss nodes provide a higher degree of exactness in comparison to an equal number of Legendre-Gauss-Lobatto nodes, a lower error of the corresponding DG approximation may be expected. In fact, higher accuracy of the DG scheme using Legendre-Gauss nodes will be experimentally shown for viscous compressible flow in Section 1.4.2. Thus, the benefit of the KEP-DG construction on interior nodes is that using Legendre-Gauss nodes instead of Legendre-Gauss-Lobatto nodes may potentially result in a more accurate approximation also for realistic problems involving viscous or inviscid compressible flow.

Moreover, we obtain the same favorable behavior regarding the KEP property for a test case of two-dimensional decaying homogeneous turbulence. More precisely, also on Legendre-Gauss nodes, the property of kinetic energy preservation of the KEP-DG scheme achieves a better representation of the expected energy spectrum.

Finally, the KEP-DG schemes are applied to the fluid equations within the moving piston

problem which represents a classical one-dimensional test case of mechanical fluid-structure interaction.

### 1.4.1 Kinetic energy preservation in one space dimension

In this section, a kinetic energy preserving DG scheme on arbitrary nodal sets with pairwise distinct nodes will be constructed. The corresponding scheme is still conservative with respect to mass, momentum and energy. In order to construct this scheme, a skew-symmetric formulation of the Euler equations of gas dynamics is used but the discretization is then related to the divergence form of the Euler equations given by

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(\rho v) = 0, \quad (1.107)$$

$$\frac{\partial}{\partial t}(\rho v) + \frac{\partial}{\partial x}(\rho v^2 + p) = 0, \quad (1.108)$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x}((\rho E + p)v) = 0, \quad (1.109)$$

for the density  $\rho$ , the velocity  $v$ , the specific total energy  $E$  and the pressure  $p$ . This system is closed by the equation of state for ideal gases  $p = (\gamma - 1)\rho(E - v^2/2)$ , with constant adiabatic coefficient  $\gamma$ .

Furthermore, a specific skew-symmetric form has been given by Morinishi in [136]. The resulting system of PDEs, which is also discretized in [61], is

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(\rho v) = 0, \quad (1.110)$$

$$\frac{1}{2} \left[ \frac{\partial}{\partial t}(\rho v) + \rho \frac{\partial}{\partial t}v \right] + \frac{1}{2} \left[ \frac{\partial}{\partial x}(\rho v^2) + \rho v \frac{\partial}{\partial x}v \right] + \frac{\partial}{\partial x}p = 0, \quad (1.111)$$

$$\frac{\partial}{\partial t}(\rho e) + \frac{\partial}{\partial x}(\rho ve + vp) - v \frac{\partial}{\partial x}p = 0, \quad (1.112)$$

where  $e$  denotes the specific inner energy and the equation of state can be rewritten as  $(\gamma - 1)\rho e = p$ . The following derivation will clarify that the first two equations, (1.110) and (1.111), are responsible for the conservation of mass and momentum as well as the correct balance of kinetic energy. The only requirement concerning the third equation (1.112) is conservation of total energy, which is already fulfilled by the standard DG discretization of the energy equation (1.109) in divergence form. Therefore, we will mainly consider the following alternative skew-symmetric form of the Euler equations,

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(\rho v) = 0, \quad (1.113)$$

$$\frac{1}{2} \left[ \frac{\partial}{\partial t}(\rho v) + \rho \frac{\partial}{\partial t}v \right] + \frac{1}{2} \left[ \frac{\partial}{\partial x}(\rho v^2) + \rho v \frac{\partial}{\partial x}v \right] + \frac{\partial}{\partial x}p = 0, \quad (1.114)$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x}(\rho vE + vp) = 0. \quad (1.115)$$

The quantities in equations (1.110)–(1.115) will be evaluated at the quadrature nodes, hence we consider the corresponding vectors of nodal values given by  $\boldsymbol{\rho} = (\rho_1, \dots, \rho_{N+1})^T$ ,  $\mathbf{v} =$

$(v_1, \dots, v_{N+1})^T$ ,  $\mathbf{e} = (e_1, \dots, e_{N+1})^T$ ,  $\mathbf{p} = (p_1, \dots, p_{N+1})^T$  and  $\mathbf{E} = (E_1, \dots, E_{N+1})^T$ . We will first directly discretize the above continuous formulations. As a second step, the discretization will be reformulated using the classical conservative variables at the quadrature nodes, denoted by

$$\begin{aligned}\mathbf{u}_1 &= (u_{1,1}, \dots, u_{1,N+1})^T \text{ with } u_{1,\nu} = \rho_\nu, \\ \mathbf{u}_2 &= (u_{2,1}, \dots, u_{2,N+1})^T \text{ with } u_{2,\nu} = \rho_\nu v_\nu, \\ \mathbf{u}_3 &= (u_{3,1}, \dots, u_{3,N+1})^T \text{ with } u_{3,\nu} = \rho_\nu e_\nu + \rho_\nu v_\nu^2 / 2,\end{aligned}$$

in order to analyze the properties of the DG scheme applied to these skew-symmetric forms. Once the point-wise values of the conservative variables are known, the point-wise values of the nodal vectors  $\mathbf{v}, \mathbf{p}, \mathbf{e}$  can be calculated by

$$v_\nu = \frac{u_{2,\nu}}{u_{1,\nu}}, \quad e_\nu = \left( \frac{u_{3,\nu}}{u_{1,\nu}} - \frac{1}{2} \frac{u_{2,\nu}^2}{u_{1,\nu}^2} \right), \quad p_\nu = (\gamma - 1) \left( u_{3,\nu} - \frac{1}{2} \frac{u_{2,\nu}^2}{u_{1,\nu}} \right), \quad \nu = 1, \dots, N + 1.$$

Furthermore, we consider the flux vector of the compressible Euler equations in conservative form and denote the vector of point-wise flux values by

$$\begin{aligned}\mathbf{f}_1 &= (f_{1,1}, \dots, f_{1,N+1})^T \text{ with } f_{1,\nu} = \rho_\nu v_\nu = u_{2,\nu}, \\ \mathbf{f}_2 &= (f_{2,1}, \dots, f_{2,N+1})^T \text{ with } f_{2,\nu} = \rho_\nu v_\nu^2 + p_\nu = v_\nu f_{1,\nu} + p_\nu, \\ \mathbf{f}_3 &= (f_{3,1}, \dots, f_{3,N+1})^T \text{ with } f_{3,\nu} = \rho_\nu v_\nu e_\nu + \rho_\nu v_\nu^3 / 2 + v_\nu p_\nu = v_\nu (u_{3,\nu} + p_\nu).\end{aligned}$$

In order to use a matrix-vector formulation in the following derivation of the kinetic energy preserving DG scheme, we need the following notation for certain diagonal matrices. For a given vector  $\mathbf{w} \in \mathbb{R}^{N+1}$  of point-wise quantities, we again denote by  $\underline{\mathbf{w}}$  the diagonal matrix

$$\underline{\mathbf{w}} = \text{diag}(w_1, \dots, w_{N+1}),$$

obtained by injecting the entries of  $\mathbf{w}$  into the diagonal.

In the following, each of the equations within the skew-symmetric formulations will be discretized by the DG scheme in one space dimension on Legendre-Gauss nodes given by the matrix-vector formulation (1.53). This basically means that any occurrence of the partial derivative  $\frac{\partial}{\partial x}$  will be substituted by the matrix  $\mathbf{D}$ . The precise choice of numerical flux functions needed within the discretization will not be specified until the kinetic energy balance is considered. For the continuity equation in divergence form given in (1.110) or (1.113) this approach obviously yields the standard DG discretization as shown below.

The discrete continuity equation is given by a direct discretization of (1.110) (or (1.113), respectively) via the DG scheme, i.e. neglecting the cell index  $i$ , we have

$$\frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_1 + \mathbf{D} \mathbf{f}_1 = \mathbf{M}^{-1} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1. \quad (1.116)$$

Thus, conservation of mass is automatically satisfied. Conservation of momentum and total energy as well as the kinetic energy balance are subject of the following discussion. The

DG discretization of equations (1.111) (or (1.114), respectively) and (1.112) will yield certain additional terms. These terms have to be chosen in a way to guarantee conservativity and consistency as stated more precisely in the following Lemma. Due to the references to interface fluxes needed both in the statement of Lemma 1.18 and in the proof, the cell index  $i$  is not neglected this time.

**Lemma 1.18.** *For a scalar conservation law denoted by  $\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}f(u(x,t)) = 0$ , we consider a cell-wise discretization of the form*

$$\begin{aligned} & \frac{\Delta x_i}{2} \frac{d}{dt} \mathbf{u}^i + \mathbf{D}\mathbf{f}^i + \left[ -\mathbf{D}\underline{\alpha}^i \boldsymbol{\beta}^i + \underline{\alpha}^i \mathbf{D} \boldsymbol{\beta}^i + \underline{\beta}^i \mathbf{D} \underline{\alpha}^i \right] \\ &= \mathbf{M}^{-1} \left( [(f_h^i - f^{*,i}) \mathbf{L}]_{-1}^1 + [\underline{\alpha}^i (\beta_h^i - \beta^{*,i}) \mathbf{L}]_{-1}^1 \right) \\ &+ \mathbf{M}^{-1} ([k_h^i \mathbf{L}]_{-1}^1 + k_i^{*,+} \mathbf{L}(-1) - k_{i+1}^{*,-} \mathbf{L}(1)), \end{aligned} \quad (1.117)$$

with arbitrary nodal values  $\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i$ , additional inner correction terms  $k_h^i$  and numerical fluxes  $k_i^{*,+}, k_{i+1}^{*,-}$  which have to be specified. In (1.117), the functions  $f^*$  and  $\beta^*$  denote locally Lipschitz continuous numerical flux functions consistent to  $f$  and  $\beta$ , respectively. Under these premises, the following assertions hold.

1. The scheme (1.117) is conservative, if and only if

$$k_h^i(-1) - k_i^{*,+} = -(\alpha\beta)_h^i(-1) + \alpha_h^i(-1)\beta_i^* - C_i^* \quad (1.118)$$

and

$$k_h^i(1) - k_{i+1}^{*,-} = -(\alpha\beta)_h^i(1) + \alpha_h^i(1)\beta_{i+1}^* - C_{i+1}^* \quad (1.119)$$

for interface-dependent values  $C_i^*, C_{i+1}^*$  depending on the nodal values  $\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{u}$  in the corresponding left and right cells.

2. If we reduce the level of data dependence for a conservative scheme and assume that

- $k_h^i$  only depends on the interior values  $\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i$ ,
- $k_i^{*,+}$  only depends on  $\beta_i^*$  and  $\alpha_h^i(-1)$ ,
- $k_{i+1}^{*,-}$  only depends on  $\beta_{i+1}^*$  and  $\alpha_h^i(1)$ ,

then we obtain  $C_i^* = C^*(\beta_i^*)$  and  $C_{i+1}^* = C^*(\beta_{i+1}^*)$  in (1.118), (1.119), i.e. the interface values of  $C^*$  only depend on the interface values of the given flux function  $\beta^*$ .

3. The scheme (1.117) with correction terms set to

$$\begin{aligned} k_h^i &= -(\alpha\beta)_h^i, \\ k_i^{*,+} &= -\alpha_h^i(-1)\beta_i^* + C^*(\beta_i^*), \\ k_{i+1}^{*,-} &= -\alpha_h^i(1)\beta_{i+1}^* + C^*(\beta_{i+1}^*), \end{aligned}$$

reduces to a consistent finite volume scheme for  $N = 0$  if and only if  $C^*(\beta_i^*) = C^*(\beta_{i+1}^*) = 0$ .

*Proof of Lemma 1.18.* For the assertion of conservativity, we multiply the scheme (1.117) from left by  $\mathbf{1}^T \mathbf{M}$ . Thus, the rate of change of the cell mean within a cell  $[x_i, x_{i+1}]$  multiplied by the cell size  $\Delta x$  is given by

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \frac{\Delta x_i}{2} \frac{d}{dt} \mathbf{u}^i &= -\mathbf{1}^T \mathbf{M} \mathbf{D} \mathbf{f}^i - \mathbf{1}^T \mathbf{M} \left[ -\mathbf{D} \underline{\alpha}^i \beta^i + \underline{\alpha}^i \mathbf{D} \beta^i + \underline{\beta}^i \mathbf{D} \alpha^i \right] \\ &\quad + \mathbf{1}^T [(f_h^i + \underline{\alpha}^i \beta_h^i + k_h^i) \mathbf{L}]_{-1}^1 - \mathbf{1}^T [f^{*,i} \mathbf{L}]_{-1}^1 - \mathbf{1}^T [\underline{\alpha}^i \beta^{*,i} \mathbf{L}]_{-1}^1 \\ &\quad + \mathbf{1}^T (k_i^{*,+} \mathbf{L}(-1) - k_{i+1}^{*,+} \mathbf{L}(1)) \\ &= -\mathbf{1}^T (\mathbf{B} - \mathbf{D}^T \mathbf{M}) \mathbf{f}^i \\ &\quad - \mathbf{1}^T \left[ -(\mathbf{B} - \mathbf{D}^T \mathbf{M}) \underline{\alpha}^i \beta^i + \underline{\alpha}^i (\mathbf{B} - \mathbf{D}^T \mathbf{M}) \beta^i + \underline{\beta}^i \mathbf{M} \mathbf{D} \alpha^i \right] \\ &\quad + \mathbf{1}^T [(f_h^i + \underline{\alpha}^i \beta_h^i + k_h^i) \mathbf{L}]_{-1}^1 - \mathbf{1}^T [f^{*,i} \mathbf{L}]_{-1}^1 - \mathbf{1}^T [\underline{\alpha}^i \beta^{*,i} \mathbf{L}]_{-1}^1 \\ &\quad + \mathbf{1}^T (k_i^{*,+} \mathbf{L}(-1) - k_{i+1}^{*,+} \mathbf{L}(1)), \end{aligned}$$

using the SBP property  $\mathbf{M} \mathbf{D} = \mathbf{B} - \mathbf{D}^T \mathbf{M}$ .

Furthermore, the interpolation of boundary values and the corresponding boundary terms are encoded in the matrix  $\mathbf{B}$ . Thus, due to Theorem 1.5 we can replace the corresponding terms by  $\mathbf{B} \mathbf{f}^i = [f_h^i \mathbf{L}]_{-1}^1$ ,  $\mathbf{B} \underline{\alpha}^i \beta^i = [(\alpha \beta)_h^i \mathbf{L}]_{-1}^1$  and  $(\alpha^i)^T \mathbf{B} \beta^i = \mathbf{1}^T \underline{\alpha}^i [\beta_h^i \mathbf{L}]_{-1}^1$ . Since this cancels out the terms containing  $f_h^i$  and  $\underline{\alpha}^i \beta_h^i$ , it holds that

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \frac{\Delta x_i}{2} \frac{d}{dt} \mathbf{u}^i &= (\mathbf{D} \mathbf{1})^T \mathbf{M} \mathbf{f}^i - [(\mathbf{D} \mathbf{1})^T \mathbf{M} \underline{\alpha}^i \beta^i - (\alpha^i)^T \mathbf{D}^T \mathbf{M} \beta^i + (\beta^i)^T \mathbf{M} \mathbf{D} \alpha^i] \\ &\quad + \mathbf{1}^T [((\alpha \beta)_h^i + k_h^i) \mathbf{L}]_{-1}^1 \\ &\quad - \mathbf{1}^T [f^{*,i} \mathbf{L}]_{-1}^1 - \mathbf{1}^T [\underline{\alpha}^i \beta^{*,i} \mathbf{L}]_{-1}^1 + \mathbf{1}^T (k_i^{*,+} \mathbf{L}(-1) - k_{i+1}^{*,+} \mathbf{L}(1)). \end{aligned}$$

Now, discrete differentiation yields  $\mathbf{D} \mathbf{1} = \mathbf{0}$ , and we obviously have  $-\alpha^T \mathbf{D}^T \mathbf{M} \beta + \beta^T \mathbf{M} \mathbf{D} \alpha = \mathbf{0}$ . This reduces the rate of change of mass contained in the specific cell to

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \frac{\Delta x_i}{2} \frac{d}{dt} \mathbf{u}^i &= \mathbf{1}^T [((\alpha \beta)_h^i + k_h^i) \mathbf{L}]_{-1}^1 - \mathbf{1}^T [f^{*,i} \mathbf{L}]_{-1}^1 - \mathbf{1}^T [\underline{\alpha}^i \beta^{*,i} \mathbf{L}]_{-1}^1 \\ &\quad + \mathbf{1}^T (k_i^{*,+} \mathbf{L}(-1) - k_{i+1}^{*,+} \mathbf{L}(1)) \end{aligned} \tag{1.120}$$

In order to guarantee the conservation property, the remaining terms may only contain interface contributions. Hence, the balance of flux contributions at cell interfaces has to be investigated using the above equation (1.120).

To this end, we consider the interface at the grid node  $x_i$  between two cells  $I_{i-1} = [x_{i-1}, x_i]$  and  $I_i = [x_i, x_{i+1}]$ . As the interpolation property yields  $\mathbf{1}^T \mathbf{L}(\xi) = 1$ , we obtain the following fluxes over this cell interface by considering (1.120) on the two adjacent cells. Neglecting the common flux  $f_i^*$  and denoting by  $C_i^{*,i-1}$  the remaining flux of mass leaving the left cell  $I_{i-1}$  from left to right through the interface  $x_i$ , we have

$$C_i^{*,i-1} = -(\alpha \beta)_h^{i-1}(1) - k_h^{i-1}(1) + \alpha_h^{i-1}(1) \beta_i^* + k_i^{*,+}, \tag{1.121}$$

whereas the analogous quantity  $C_i^{*,i}$  of mass entering the right cell  $I_i$  through  $x_i$  is given by

$$C_i^{*,i} = -(\alpha\beta)_h^i(-1) - k_h^i(-1) + \alpha_h^i(-1)\beta_i^* + k_i^{*,+}. \quad (1.122)$$

Now, the conservation property precisely requires the equality

$$C_i^{*,i-1} = C_i^{*,i} =: C_i^*, \quad (1.123)$$

yielding the first assertion.

As for the second assertion, we now assume that  $k_h^i$  only depends on the interior values  $\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i$  and that  $k_i^{*,+}$  only depends on  $\beta_i^*$  and  $\alpha_h^i(-1)$ . If we then modify any of the input values other than  $\boldsymbol{\alpha}^i, \boldsymbol{\beta}^i, \beta_i^*$ , the value of  $C_i^{*,i}$  in (1.122) does not change and thus  $C_i^*$  remains constant as well due to the equality in (1.123). If we furthermore assume that  $k_i^{*,-}$  only depends on  $\beta_i^*$  and  $\alpha_h^{i-1}(1)$ , modifying  $\boldsymbol{\alpha}^i$  or  $\boldsymbol{\beta}^i$  while leaving  $\beta_i^*$  constant does not change  $C_i^{*,i-1}$  in (1.121) and hence does not change  $C_i^*$  due to (1.123). Therefore,  $C_i^*$  only depends on  $\beta_i^*$  which proves assertion 2.

Now, consistency in the finite volume sense refers to a consistent numerical flux. With  $k_h^i, k_i^{*,+}, k_{i+1}^{*,-}$  set as in the third assertion, the scheme (1.117) for  $N = 0$  reduces to

$$\Delta x_i \frac{d}{dt} u^i = [f^{*,i}]_{-1}^1 - [\alpha^i \beta^{*,i}]_{-1}^1 + (k_i^{*,+} - k_{i+1}^{*,-}) = [f^{*,i}]_{-1}^1 - (C_{i+1}^* - C_i^*).$$

Since  $f^*$  is consistent to  $f$  with  $f^*(u, u) = f(u)$  and  $\beta^*$  is consistent to  $\beta$ , this above finite volume scheme is consistent if and only if

$$f^*(u, u) + C^*(\beta^*(u, u)) = f(u) + C^*(\beta(u)) = f(u).$$

This results in the requirement  $C^*(\beta(u)) = 0$ , which needs to hold for any cell interface and any value of the conserved variable  $u$ . Therefore, the last assertion is proven as well.  $\square$

### The discrete momentum equation

In the following derivations the skew-symmetric formulation of the momentum equation will be discretized by the DG scheme using Legendre-Gauss nodes. The available freedom in choosing the numerical flux function will be used to obtain a scheme which conserves momentum and additionally fulfills a kinetic energy balance with respect to the cell means of kinetic energy. This part follows closely the Legendre-Gauss-Lobatto case in [61].

A direct discretization of the momentum equation, neglecting the cell index  $i$ , yields

$$\frac{\Delta x}{2} \frac{1}{2} \left[ \frac{d}{dt} \mathbf{u}_2 + \underline{\mathbf{u}}_1 \frac{d}{dt} \mathbf{v} \right] + \frac{1}{2} \left[ \mathbf{D} \underline{\rho} \mathbf{v}^2 + \underline{\rho} \underline{\mathbf{v}} \mathbf{D} \mathbf{v} \right] + \mathbf{D} \mathbf{p} = \mathbf{M}^{-1} [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1, \quad (1.124)$$

with terms  $g_{2,h}$  and  $g_2^*$  which are not yet specified. We will choose these terms in a manner to guarantee conservation of momentum. Therefore, we multiply (1.116) from left by  $\frac{1}{2} \underline{\mathbf{v}}$  and add the resulting equation to (1.124). Due to continuity in time, this yields

$$\begin{aligned} & \frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_2 + \mathbf{D} (\underline{\rho} \mathbf{v}^2 + \mathbf{p}) + \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^2 + \underline{\rho} \underline{\mathbf{v}} \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \mathbf{f}_1 \right] \\ &= \mathbf{M}^{-1} \left( [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 + \frac{1}{2} \underline{\mathbf{v}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \right). \end{aligned} \quad (1.125)$$

Setting  $\boldsymbol{\alpha} = \frac{1}{2}\mathbf{v}$ ,  $\boldsymbol{\beta} = \underline{\underline{\rho}}\mathbf{v} = \mathbf{f}_1$  and  $\beta^* = f_1^*$ , due to Lemma 1.18, we obtain a consistent and conservative scheme if we set

$$g_{2,h} = f_{2,h} + k_{2,h} = f_{2,h} - (\alpha\beta)_h = \left( \frac{1}{2}\rho v^2 + p \right)_h \quad (1.126)$$

and

$$g_2^*(-1) = f_2^* - \frac{1}{2}v^+ f_1^*, \quad g_2^*(1) = f_2^* - \frac{1}{2}v^- f_1^*,$$

where  $v^+ = v_h(-1)$ ,  $v^- = v_h(1)$  denote the one-sided limits of  $v_h$  at the cell interfaces. The formulation (1.125) can finally be rewritten as

$$\frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_2 + \mathbf{D} \mathbf{f}_2 + \mathbf{s}_2 = \mathbf{M}^{-1} [(f_{2,h} - f_2^*) \mathbf{L}]_{-1}^1 + \mathbf{M}^{-1} \mathbf{s}_2^{bc}, \quad (1.127)$$

with the volume terms

$$\mathbf{s}_2 = \frac{1}{2} \left[ -\mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{v} + \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v} \right]$$

and the boundary correction

$$\mathbf{s}_2^{bc} = \frac{1}{2} (\underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 - [((\rho v^2)_h - v^\pm f_1^*) \mathbf{L}]_{-1}^1).$$

We may as well derive a weak formulation of this skew-symmetric discretization. Using the SBP property  $\mathbf{MD} = \mathbf{B} - \mathbf{D}^T \mathbf{M}$  with  $\mathbf{Bf} = [f_h \mathbf{L}]_{-1}^1$  as well as the definition of  $g_{2,h}$  and  $f_{2,h} = (\rho v^2 + p)_h$ , equation (1.125) results in

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{Mu}_2 &= \mathbf{D}^T \mathbf{M} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 + \mathbf{p} \right) - \frac{1}{2} \left[ \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}} \mathbf{MDv} - \underline{\underline{\mathbf{v}}} \mathbf{D}^T \mathbf{Mf}_1 \right] \\ &\quad - \mathbf{B} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 + \mathbf{p} \right) - \frac{1}{2} \underline{\underline{\mathbf{v}}} \mathbf{Bf}_1 + \left( [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 + \frac{1}{2} \underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \right), \end{aligned}$$

which is equivalent to the weak formulation

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{Mu}_2 &= \mathbf{D}^T \mathbf{M} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 + \mathbf{p} \right) - \frac{1}{2} \left[ \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}} \mathbf{MDv} - \underline{\underline{\mathbf{v}}} \mathbf{D}^T \mathbf{Mf}_1 \right] - [g_2^* \mathbf{L}]_{-1}^1 - \frac{1}{2} \underline{\underline{\mathbf{v}}} [f_1^* \mathbf{L}]_{-1}^1 \\ &= \mathbf{S}^T \mathbf{g}_2 - \frac{1}{2} \left[ \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}} \mathbf{Sv} - \underline{\underline{\mathbf{v}}} \mathbf{S}^T \mathbf{f}_1 \right] - [f_2^* \mathbf{L}]_{-1}^1 + \frac{1}{2} [(v^\pm \mathbf{I} - \underline{\underline{\mathbf{v}}}) f_1^* \mathbf{L}]_{-1}^1. \end{aligned} \quad (1.128)$$

In fact, for efficiency reasons, this weak formulation should be implemented instead of the strong form as it does not require boundary interpolation of the flux values  $f_1$  and  $g_2$ .

### The discrete kinetic energy balance

Before considering the energy equation, i.e. the third equation of the system of Euler equations, we will derive the kinetic energy balance of the scheme. This will lead to additional constraints on the numerical flux function  $\mathbf{f}^*$ .

First, we multiply the discrete momentum equation (1.124) with  $\underline{\underline{\mathbf{v}}}$  and obtain

$$\frac{\Delta x}{2} \frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \frac{d}{dt} \mathbf{u}_2 + \underline{\underline{\mathbf{u}}}_2 \frac{d}{dt} \mathbf{v} \right] + \frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} \right] + \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{p} = \mathbf{M}^{-1} \underline{\underline{\mathbf{v}}} [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1,$$

where we used the fact that  $\mathbf{M}$  is diagonal.

Time derivatives can be recast to

$$\frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \frac{d}{dt} \mathbf{u}_2 + \underline{\underline{\mathbf{u}}}_2 \frac{d}{dt} \mathbf{v} \right] = \frac{d}{dt} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 \right)$$

hence, assuming time continuity, the kinetic energy balance is given by

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{e}_{kin} &= \frac{\Delta x}{2} \frac{d}{dt} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 \right) \\ &= -\frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} \right] - \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{p} + \mathbf{M}^{-1} \underline{\underline{\mathbf{v}}} [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1. \end{aligned} \quad (1.129)$$

The balance of the cell means of kinetic energy is then obtained by multiplying the above equation by  $\mathbf{1}^T \mathbf{M}$ . That is, the total amount of kinetic energy within the cell is subject to the following rate of change

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{1}^T \mathbf{M} \mathbf{e}_{kin} &= -\frac{1}{2} \left[ \mathbf{v}^T \mathbf{M} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + (\underline{\underline{\rho}} \mathbf{v}^2)^T \mathbf{M} \mathbf{D} \mathbf{v} \right] - \mathbf{v}^T \mathbf{M} \mathbf{D} \mathbf{p} + \mathbf{v}^T [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 \\ &= -\frac{1}{2} \left[ \mathbf{v}^T \mathbf{B} \underline{\underline{\rho}} \mathbf{v}^2 - (\mathbf{D} \mathbf{v})^T \mathbf{M} \underline{\underline{\rho}} \mathbf{v}^2 + (\underline{\underline{\rho}} \mathbf{v}^2)^T \mathbf{M} \mathbf{D} \mathbf{v} \right] \\ &\quad - \mathbf{v}^T \mathbf{B} \mathbf{p} + (\mathbf{D} \mathbf{v})^T \mathbf{M} \mathbf{p} + \mathbf{v}^T [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 \\ &= -\mathbf{v}^T \mathbf{B} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 + \mathbf{p} \right) + (\mathbf{D} \mathbf{v})^T \mathbf{M} \mathbf{p} + \mathbf{v}^T [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1, \end{aligned}$$

where the SBP property  $\mathbf{M} \mathbf{D} = \mathbf{B} - \mathbf{D}^T \mathbf{M}$  as well as the equality

$$-(\mathbf{D} \mathbf{v})^T \mathbf{M} \underline{\underline{\rho}} \mathbf{v}^2 + (\underline{\underline{\rho}} \mathbf{v}^2)^T \mathbf{M} \mathbf{D} \mathbf{v} = 0$$

were used. Furthermore, since  $\mathbf{v}^T \mathbf{B} \mathbf{g}_2 = \mathbf{v}^T [g_{2,h} \mathbf{L}]_{-1}^1$  by Theorem 1.5, we obtain

$$\begin{aligned} \frac{d}{dt} \mathbf{1}^T \mathbf{M} \mathbf{e}_{kin} &= -\mathbf{v}^T \mathbf{B} \mathbf{g}_2 + (\mathbf{D} \mathbf{v})^T \mathbf{M} \mathbf{p} + \mathbf{v}^T [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 \\ &= (\mathbf{D} \mathbf{v})^T \mathbf{M} \mathbf{p} - \mathbf{v}^T [g_2^* \mathbf{L}]_{-1}^1. \end{aligned} \quad (1.130)$$

The volume term  $(\mathbf{D} \mathbf{v})^T \mathbf{M} \mathbf{p}$  represents a change of kinetic energy due to a pressure variation which is physically correct. Hence, only the contribution containing the auxiliary flux  $g_2^*$  remains to be dealt with. For a correct balance of kinetic energy, only the terms contained in  $g_2^*$  which are related to transport have to vanish. In this regard, we split  $g_2^{*,\pm} = f_2^* - \frac{1}{2} v^\pm f_1^*$  into a transport and a pressure term,  $g_2^{*,\pm} = \tilde{f}_2^* + p^* - \frac{1}{2} v^\pm f_1^* = \tilde{g}_2^{*,\pm} + p^*$  as in [61]. For the contribution of interface fluxes at the interface  $(i-1, i)$  to the cell means of kinetic energy as in (1.130) we then demand

$$(\mathbf{v}^i)^T \tilde{g}_2^{*,\pm} \mathbf{L}(-1) = (\mathbf{v}^{i-1})^T \tilde{g}_2^{*,\pm} \mathbf{L}(1).$$

This yields the condition

$$\begin{aligned} 0 &= \left( \tilde{f}_2^* - \frac{1}{2} v^+ f_1^* \right) (\mathbf{v}^i)^T \mathbf{L}(-1) - \left( \tilde{f}_2^* - \frac{1}{2} v^- f_1^* \right) (\mathbf{v}^{i-1})^T \mathbf{L}(1) \\ &= \left( \tilde{f}_2^* - \frac{1}{2} v^+ f_1^* \right) v^+ - \left( \tilde{f}_2^* - \frac{1}{2} v^- f_1^* \right) v^- \\ &= \tilde{f}_2^* (v^+ - v^-) + \frac{1}{2} ((v^-)^2 - (v^+)^2) f_1^*, \end{aligned}$$

thus, the numerical flux  $\tilde{f}_2^*$  corresponding to the transport part  $\rho v^2$  of  $f_2$  is required to fulfill

$$\tilde{f}_2^* = \frac{v^+ + v^-}{2} f_1^*. \quad (1.131)$$

With this property of the numerical flux function, we have

$$g_2^{*,\pm} = \tilde{f}_2^* + p^* - \frac{1}{2} v f_1^* = \frac{v^+ + v^-}{2} f_1^* + p^* - \frac{1}{2} v^\pm f_1^* = \frac{1}{2} v^\mp f_1^* + p^* \quad (1.132)$$

and the condition

$$f_2^* = \tilde{f}_2^* + p^* = \frac{v^+ + v^-}{2} f_1^* + p^*. \quad (1.133)$$

There are many possibilities to construct numerical flux functions satisfying (1.133). Here, we will consider the following numerical flux functions.

- The so-called KEP flux  $\mathbf{f}_A^* = \mathbf{f}_{KEP}^*$  by Jameson, e.g. given in [92],

$$\begin{aligned} f_{A,1}^* &= \bar{\rho} \bar{v}, \\ f_{A,2}^* &= \bar{\rho} \bar{v}^2 + \bar{p}, \\ f_{A,3}^* &= \bar{\rho} \bar{v} \bar{H}, \quad H = E + \frac{p}{\rho}, \end{aligned} \quad (1.134)$$

where the quantity  $\bar{q}$  denotes the arithmetic average  $\bar{q} = \frac{1}{2}(q^+ + q^-)$  for given left and right values  $q^\pm$ .

- The kinetic energy preserving and entropy consistent numerical flux given in [39], which will be denoted KEP-EC in this work,

$$\begin{aligned} f_{B,1}^* &= \hat{\rho} \bar{v}, \\ f_{B,2}^* &= \frac{\bar{\rho}}{2\beta} + \bar{v} f_{B,1}^*, \\ f_{B,3}^* &= \frac{1}{2(\gamma-1)\hat{\beta}} - \frac{\bar{v}^2}{2} f_{B,1}^* + \bar{v} f_{B,2}^*, \end{aligned} \quad (1.135)$$

where  $\beta = \frac{\rho}{2p}$  and, given left and right values  $q^\pm$ , the quantity  $\hat{q}$  denotes the logarithmic average

$$\hat{q} = \frac{q^- - q^+}{\ln(q^-) - \ln(q^+)}. \quad (1.136)$$

For  $q^- \approx q^+$  the numerically stable approximation to (1.136) given in [89] will be used.

- A modified version of the classical van Leer flux  $\mathbf{f}_{VL}^*$  given in [108] specifically designed to fulfill the KEP property (1.133). This flux  $\mathbf{f}_C^*$  is denoted by KEP-VL and given by

$$\begin{aligned} f_{C,1}^* &= f_{VL,1}^*, \\ f_{C,2}^* &= \bar{v} f_{VL,1}^* + \bar{p}, \\ f_{C,3}^* &= f_{VL,3}^*. \end{aligned} \quad (1.137)$$

### The discrete energy equation

In the previous paragraphs we constructed a skew-symmetric DG discretization of the continuity and momentum equations which is i) mass and momentum conserving and ii) fulfills a balance of kinetic energy. Moreover, a direct discretization of the divergence form (1.109) accomplishes the task of total energy conservation. The resulting conservative scheme then exhibits all the desired properties and will be numerically tested in Section 1.4.2.

In addition, as in [61], a conservative discretization may be obtained for the alternative skew symmetric form (1.112). Again, the derivations for the DG scheme on Legendre-Gauss nodes leads to modifications of the skew-symmetric terms with respect to the interface contributions. First, the discretization of equation (1.112) takes on a more general form given by

$$\frac{\Delta x}{2} \frac{d}{dt} (\underline{\underline{\rho}} \mathbf{e}) + \mathbf{D}(\underline{\underline{\rho}} \mathbf{v} \mathbf{e} + \underline{\underline{\mathbf{v}}} \mathbf{p}) - \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{p} = \mathbf{M}^{-1} [(G_{3,h} - G_3^*) \mathbf{L}]_{-1}^1, \quad (1.138)$$

where  $G_{3,h}$  and  $G_3^*$  now denote matrix valued quantities which have to be specified.

Using the discrete kinetic energy balance (1.129) in the form

$$-\underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{p} = \frac{\Delta x}{2} \frac{d}{dt} \left( \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2 \right) + \frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} \right] - \mathbf{M}^{-1} \underline{\underline{\mathbf{v}}} [(g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1,$$

the corresponding term  $-\underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{p}$  in (1.138) may be substituted.

Hence, for the total energy  $\mathbf{u}_3 = (\underline{\underline{\rho}} \mathbf{e} + \frac{1}{2} \underline{\underline{\rho}} \mathbf{v}^2)$  we have

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_3 + \mathbf{D}(\underline{\underline{\rho}} \mathbf{v} \mathbf{e} + \underline{\underline{\mathbf{v}}} \mathbf{p}) + \frac{1}{2} \left[ \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} \right] \\ = \mathbf{M}^{-1} [(G_{3,h} - G_3^*) \mathbf{L} + \underline{\underline{\mathbf{v}}} (g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1, \end{aligned}$$

which can be rearranged to

$$\begin{aligned} \frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_3 + \mathbf{D} \mathbf{f}_3 + \frac{1}{2} \left[ -\mathbf{D} \underline{\underline{\rho}} \mathbf{v}^3 + \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} \right] \\ = \mathbf{M}^{-1} [(G_{3,h} - G_3^*) \mathbf{L} + \underline{\underline{\mathbf{v}}} (g_{2,h} - g_2^*) \mathbf{L}]_{-1}^1 \\ = \mathbf{M}^{-1} \left[ (G_{3,h} - G_3^*) \mathbf{L} + \underline{\underline{\mathbf{v}}} (p_h - (p^* + \frac{1}{2} (\tilde{f}_2^* - v^\pm f_1^*))) \mathbf{L} + \frac{1}{2} \underline{\underline{\mathbf{v}}} ((\rho v^2)_h - \tilde{f}_2^*) \mathbf{L} \right]_{-1}^1, \end{aligned} \quad (1.139)$$

using the representations of  $g_{2,h}$  and  $g_2^*$  as in (1.126) and (1.132), respectively.

Also in this case, Lemma 1.18 can be applied again to (1.139) using  $\boldsymbol{\alpha} = \mathbf{v}$ ,  $\boldsymbol{\beta} = \frac{1}{2}\underline{\underline{\rho}}\mathbf{v}^2$  and  $\beta^* = \frac{1}{2}\tilde{f}_2^*$ . A conservative scheme is then obtained by setting

$$G_{3,h} + \underline{\underline{\mathbf{v}}}p_h = (f_{3,h} - (\alpha\beta)_h)\mathbf{I} = (\rho v e + vp)_h \mathbf{I},$$

where  $\mathbf{I}$  denotes the identity matrix, and

$$G_3^{*,\pm} + \underline{\underline{\mathbf{v}}} \left( p^* + \frac{1}{2}(\tilde{f}_2^* - v^\pm f_1^*) \right) = (f_3^* - \alpha^\pm \beta^*)\mathbf{I} = \left( f_3^* - \frac{1}{2}v^\pm \tilde{f}_2^* \right) \mathbf{I},$$

This yields

$$G_{3,h} + \underline{\underline{\mathbf{v}}}p_h + \frac{1}{2}\underline{\underline{\mathbf{v}}}(\rho v^2)_h = f_{3,h}\mathbf{I} - \frac{1}{2}(\rho v^3)_h \mathbf{I} + \frac{1}{2}\underline{\underline{\mathbf{v}}}(\rho v^2)_h$$

and

$$G_3^{*,\pm} + \underline{\underline{\mathbf{v}}} \left( p^* + \frac{1}{2}(\tilde{f}_2^* - v^\pm f_1^*) + \frac{1}{2}\tilde{f}_2^* \right) = \left( f_3^* - \frac{1}{2}v^\pm \tilde{f}_2^* \right) \mathbf{I} + \frac{1}{2}\underline{\underline{\mathbf{v}}}\tilde{f}_2^*.$$

Hence, the formulation (1.139) results in

$$\frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_3 + \mathbf{D} \mathbf{f}_3 + \mathbf{s}_3 = \mathbf{M}^{-1}[(f_{3,h} - f_3^*)\mathbf{L}]_{-1}^1 + \mathbf{M}^{-1}\mathbf{s}_3^{bc}, \quad (1.140)$$

with the volume terms

$$\mathbf{s}_3 = \frac{1}{2} \left[ -\mathbf{D} \underline{\underline{\rho}} \mathbf{v}^3 + \underline{\underline{\rho}} \underline{\underline{\mathbf{v}}}^2 \mathbf{D} \mathbf{v} + \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\rho}} \mathbf{v}^2 \right]$$

and the boundary correction

$$\mathbf{s}_3^{bc} = \frac{1}{2} \left( \underline{\underline{\mathbf{v}}} [((\rho v^2)_h - \tilde{f}_2^*)\mathbf{L}]_{-1}^1 - [((\rho v^3)_h - v^\pm \tilde{f}_2^*)\mathbf{L}]_{-1}^1 \right),$$

with  $\tilde{f}_2^*$  as in (1.131).

### Summary of skew-symmetric DG discretizations

For convenience, we summarize the alternative discretizations of the Euler equations at this point. The resulting discrete continuity, momentum and energy equation have the general form

$$\frac{\Delta x}{2} \frac{d}{dt} \mathbf{u}_k + \mathbf{D} \mathbf{f}_k + \mathbf{s}_k = \mathbf{M}^{-1}[(f_{k,h} - f_k^*)\mathbf{L}]_{-1}^1 + \mathbf{M}^{-1}\mathbf{s}_k^{bc}, \quad k = 1, 2, 3.$$

With respect to the following numerical investigations, four alternative choices for the volume terms  $\mathbf{s}_k$  and boundary corrections  $\mathbf{s}_k^{bc}$  may be considered, i.e. the standard DG scheme which is conservative regarding mass, momentum and energy, two skew-symmetric forms which additionally preserve the kinetic energy balance as well as a naive application of the kinetic energy preserving DG scheme in [61] to Legendre-Gauss nodes. That is, we have the following specifications.

1. The standard DG discretization is given by

$$\mathbf{s}_k = \mathbf{s}_k^{bc} = \mathbf{0}, \quad k = 1, 2, 3. \quad (1.141)$$

2. The skew-symmetric DG discretization based on Morinishi's skew-symmetric form,

$$\begin{aligned}
\mathbf{s}_1 &= \mathbf{0}, \\
\mathbf{s}_2 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^2 + \underline{\rho} \underline{\mathbf{v}} \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \mathbf{v} \right], \\
\mathbf{s}_3 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^3 + \underline{\rho} \underline{\mathbf{v}}^2 \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \mathbf{v}^2 \right], \\
\mathbf{s}_1^{bc} &= \mathbf{0}, \\
\mathbf{s}_2^{bc} &= \frac{1}{2} \left( \underline{\mathbf{v}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 - [((\rho v^2)_h - v^\pm f_1^*) \mathbf{L}]_{-1}^1 \right), \\
\mathbf{s}_3^{bc} &= \frac{1}{2} \left( \underline{\mathbf{v}} [((\rho v^2)_h - \tilde{f}_2^*) \mathbf{L}]_{-1}^1 - [((\rho v^3)_h - v^\pm \tilde{f}_2^*) \mathbf{L}]_{-1}^1 \right)
\end{aligned}$$

with  $\tilde{f}_2^*$  as in (1.131).

3. The skew-symmetric DG discretization based on the first two equations of Morinishi's skew-symmetric form, (1.110), (1.111), as well as the energy equation in divergence form (1.109). This alternative formulation is given by

$$\begin{aligned}
\mathbf{s}_1 &= \mathbf{s}_3 = \mathbf{0}, \\
\mathbf{s}_2 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^2 + \underline{\rho} \underline{\mathbf{v}} \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \mathbf{v} \right], \\
\mathbf{s}_1^{bc} &= \mathbf{s}_3^{bc} = \mathbf{0}, \\
\mathbf{s}_2^{bc} &= \frac{1}{2} \left( \underline{\mathbf{v}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 - [((\rho v^2)_h - v^\pm f_1^*) \mathbf{L}]_{-1}^1 \right).
\end{aligned} \tag{1.142}$$

4. An additional formulation is obtained, when the DG discretization on Legendre-Gauss-Lobatto nodes in [61] is naively transferred to the Legendre-Gauss case. Then we have

$$\begin{aligned}
\mathbf{s}_1 &= \mathbf{0}, \\
\mathbf{s}_2 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^2 + \underline{\rho} \underline{\mathbf{v}} \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \mathbf{v} \right], \\
\mathbf{s}_3 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \mathbf{v}^3 + \underline{\rho} \underline{\mathbf{v}}^2 \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \mathbf{v}^2 \right], \\
\mathbf{s}_k^{bc} &= \mathbf{0}, \quad k = 1, 2, 3.
\end{aligned} \tag{1.143}$$

However, in this case, the boundary treatment is inconsistent to the skew-symmetric terms if Legendre-Gauss nodes are used. This will result in stability problems as shown in Section 1.4.2.

These alternative discretizations are completed by the choice of a numerical flux function  $\mathbf{f}^*$ . If the kinetic energy balance is to be preserved, one of the numerical fluxes given in the previous paragraph, satisfying the KEP condition (1.133), will be chosen.

### 1.4.2 Numerical experiments: accuracy and evolution of kinetic energy

In the following, numerical results are reported regarding the comparison of the DG discretizations with or without skew-symmetric terms, i.e. (1.141), (1.142) and (1.143), which are furthermore equipped with the kinetic energy preserving numerical fluxes given in the previous Section 1.4.1 in addition to the classical van Leer flux [108]. For time discretization, the classical fourth order Runge-Kutta scheme is used. In all test cases, the adiabatic coefficient is set to  $\gamma = 1.4$ . Considering notation, we reserve lower case bold letters such as the vector  $\mathbf{u}$  for the nodal values of the DG scheme, while we use  $\mathbf{U}$  for the exact solution in the case of systems such as the Euler equations of gas dynamics and upper case letters  $\mathbf{F}$  and  $\mathbf{Q}$  for flux functions and source terms, respectively.

#### Experimental order of convergence

The first test case numerically investigates the order of convergence of the kinetic energy preserving scheme based on classical Legendre-Gauss nodes. As in [61], a manufactured solution is used to test the order of the scheme. For this purpose, the Euler equations (1.107),(1.108), (1.109) are augmented by a source term  $\mathbf{Q}(x, t)$ . More precisely, we consider the exact solution

$$\mathbf{U}_{ms}(x, t) = \begin{pmatrix} u(x, t) \\ u(x, t) \\ u^2(x, t) \end{pmatrix}, \quad u(x, t) = 2 + 0.1 \sin(2\pi(x - t)),$$

of the balance law

$$\frac{\partial}{\partial t} \mathbf{U}(x, t) + \frac{\partial}{\partial x} \mathbf{F}(\mathbf{U}(x, t)) = \mathbf{Q}(x, t), \quad \mathbf{U} = \begin{pmatrix} \rho \\ \rho v \\ \rho E \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(\rho E + p) \end{pmatrix}, \quad (1.144)$$

where the source term is given by

$$\mathbf{Q}(x, t) = \begin{pmatrix} 0 \\ p(x, t) \\ p(x, t) \end{pmatrix}, \quad p(x, t) = 0.28\pi \cos(2\pi(x - t)) + 0.008\pi \sin(4\pi(x - t)).$$

Thus, the source term is specifically designed to enforce  $\mathbf{U}_{ms}(x, t)$  as the exact solution of the continuous system of equations. The initial conditions on the computational domain  $\Omega = [0, 1]$  are given by  $\mathbf{U}(x, 0) = \mathbf{U}_{ms}(x, 0)$  and periodic boundary conditions are chosen. Tables 1.2, 1.3 and 1.4 list the  $L^2$  errors and corresponding experimental order of convergence obtained by the DG scheme with skew symmetric terms (1.142) for polynomial degrees  $N = 2, 3$  and  $N = 4$ , respectively, using the different numerical flux functions given in Section 1.4.1 as well as the classical van Leer flux for reference. The number of grid cells the computational domain  $\Omega$  is divided into is denoted by  $K$ . All computations were carried out until  $t_{end} = 10$  with time steps small enough in order to make temporal errors negligible. Similar to the results in [61], the scheme using central numerical fluxes which disregard upwind information show an order reduction for odd polynomial degrees, i.e. for  $N = 3$  the order is reduced to  $EOC = 3$  instead of  $EOC = N + 1$  in the even cases  $N = 2, 4$ . Hence, this observation made

$K$	van Leer		KEP (1.134)		KEP-EC (1.135)		KEP-VL (1.137)	
	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC
10	7.23e-05	-	8.66e-05	-	8.68e-05	-	1.57e-03	-
20	5.95e-06	3.60	4.34e-06	4.32	4.33e-06	4.32	1.15e-04	3.77
40	6.64e-07	3.16	4.26e-07	3.35	4.26e-07	3.35	8.33e-06	3.78
80	8.18e-08	3.02	5.26e-08	3.02	5.26e-08	3.02	8.35e-07	3.32

Table 1.2:  $L^2$  errors and experimental order of convergence (EOC) of the skew-symmetric Legendre-Gauss DG scheme, case  $N = 2$ , using different numerical fluxes.

$K$	van Leer		KEP (1.134)		KEP-EC (1.135)		KEP-VL (1.137)	
	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC
10	3.82e-06	-	2.72e-05	-	2.64e-05	-	1.73e-04	-
20	1.36e-07	4.81	4.35e-06	2.64	4.30e-06	2.62	1.53e-05	3.50
40	6.35e-09	4.42	5.20e-07	3.06	5.19e-07	3.05	1.81e-06	3.07
80	3.91e-10	4.02	6.38e-08	3.03	6.38e-08	3.02	2.29e-07	2.98

Table 1.3:  $L^2$  errors and experimental order of convergence (EOC) of the skew-symmetric Legendre-Gauss DG scheme, case  $N = 3$ , using different numerical fluxes.

in [61] can be attested also in the case of Legendre-Gauss nodes. In a direct comparison of the numerical fluxes used in the DG scheme for a constant polynomial degree, the KEP-VL flux (1.137) yields the largest errors while the original van Leer flux performs best in this setting. Hence, preserving the kinetic energy is no guarantee for better accuracy. Rather, kinetic energy preservation is a property which mimics a qualitative behavior of the exact solution. In fact, preservation of kinetic energy itself by different variants of the DG scheme has to be studied more carefully. This is the purpose of the following test case.

$K$	van Leer		KEP (1.134)		KEP-EC (1.135)		KEP-VL (1.137)	
	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC	$L^2$ error	EOC
10	1.11e-07	-	7.61e-08	-	7.61e-08	-	2.05e-06	-
20	2.21e-09	5.65	1.34e-09	5.82	1.34e-09	5.82	4.72e-08	5.44
40	4.91e-11	5.49	3.26e-11	5.37	3.26e-11	5.37	6.71e-10	6.14
80	1.51e-12	5.02	1.01e-12	5.01	1.01e-12	5.01	1.77e-11	5.24

Table 1.4:  $L^2$  errors and experimental order of convergence (EOC) of the skew-symmetric Legendre-Gauss DG scheme, case  $N = 4$ , using different numerical fluxes.

### Conservation of mean kinetic energy

In the following, we consider a special set-up to study the conservation of kinetic energy. Measuring the KEP property of a scheme is not straightforward as kinetic energy is generally not conserved in the exact solution but in balance with the term  $\underline{\mathbf{v}} \mathbf{D} \mathbf{p}$ , see equation (1.129). In this second test we therefore neglect the pressure term in the Euler equations, i.e. we consider the case of constant pressure. As a result, the energy equation is automatically fulfilled and can be dropped from the system of equations. Of course, to remain with a consistent formulation, the pressure is also neglected in the numerical flux function. Under these assumptions as specified above, the Euler equations reduce to the system

$$\begin{aligned}\frac{\partial}{\partial t} \rho + \frac{\partial}{\partial x} (\rho v) &= 0, \\ \frac{\partial}{\partial t} (\rho v) + \frac{\partial}{\partial x} (\rho v^2) &= 0.\end{aligned}$$

The components of the corresponding numerical flux function are given by

$$f_1^* = (\rho v)^* = \bar{\rho} \bar{v}, \quad f_2^* = (\rho v^2)^* = \bar{\rho} \bar{v}^2 = \bar{v} f_1^*,$$

which is precisely the reduction of the KEP flux (1.134) to the reduced system with fluxes

$$f_1 = \rho v, \quad f_2 = \rho v^2.$$

Initial conditions are given by the initial density and velocity distributions

$$\rho(x, 0) = 2, \quad v(x, 0) = \cos(2\pi x),$$

on the computational domain  $\Omega = [0, 1]$  with periodic boundary conditions.

First, Figure 1.2a depicts the time evolution of the mean kinetic energy  $\bar{e}_{kin}(t)$ , given by

$$\bar{e}_{kin}(t) = \sum_i \frac{\Delta x_i}{2} \mathbf{1}^T \mathbf{M} \mathbf{e}_{kin}(t) = \sum_i \int_{x_i}^{x_{i+1}} \left( \frac{1}{2} \rho v^2 \right)_h(t) dx,$$

for different variants of the skew symmetric terms at the end of Section 1.4.1 in the case  $N = 1$ . Here, we compare the DG scheme with the correctly derived skew-symmetric terms and boundary treatment (1.142), denoted by 'DG skew1', the standard DG scheme (1.141) without skew-symmetric terms as well as the skew symmetric terms with inconsistent boundary treatment as in (1.143), denoted by 'DG skew2'. We may observe that only the Legendre-Gauss 'DG skew1' scheme preserves the mean kinetic energy while the standard DG scheme dissipates this quantity. Skew symmetric terms with inconsistent boundary treatment for 'DG skew2' lead to an non-physical increase of kinetic energy. This increase of kinetic energy leads to oscillations of the DG solution, visible in Figure 1.2b showing the distribution of kinetic energy for the different variants of the DG scheme at output time  $t_{end} = 0.12$ . In the density distribution depicted Figure 1.2c this effect is not present but it can be observed in terms of less pronounced oscillations for the velocity distribution shown in Fig. 1.2d.

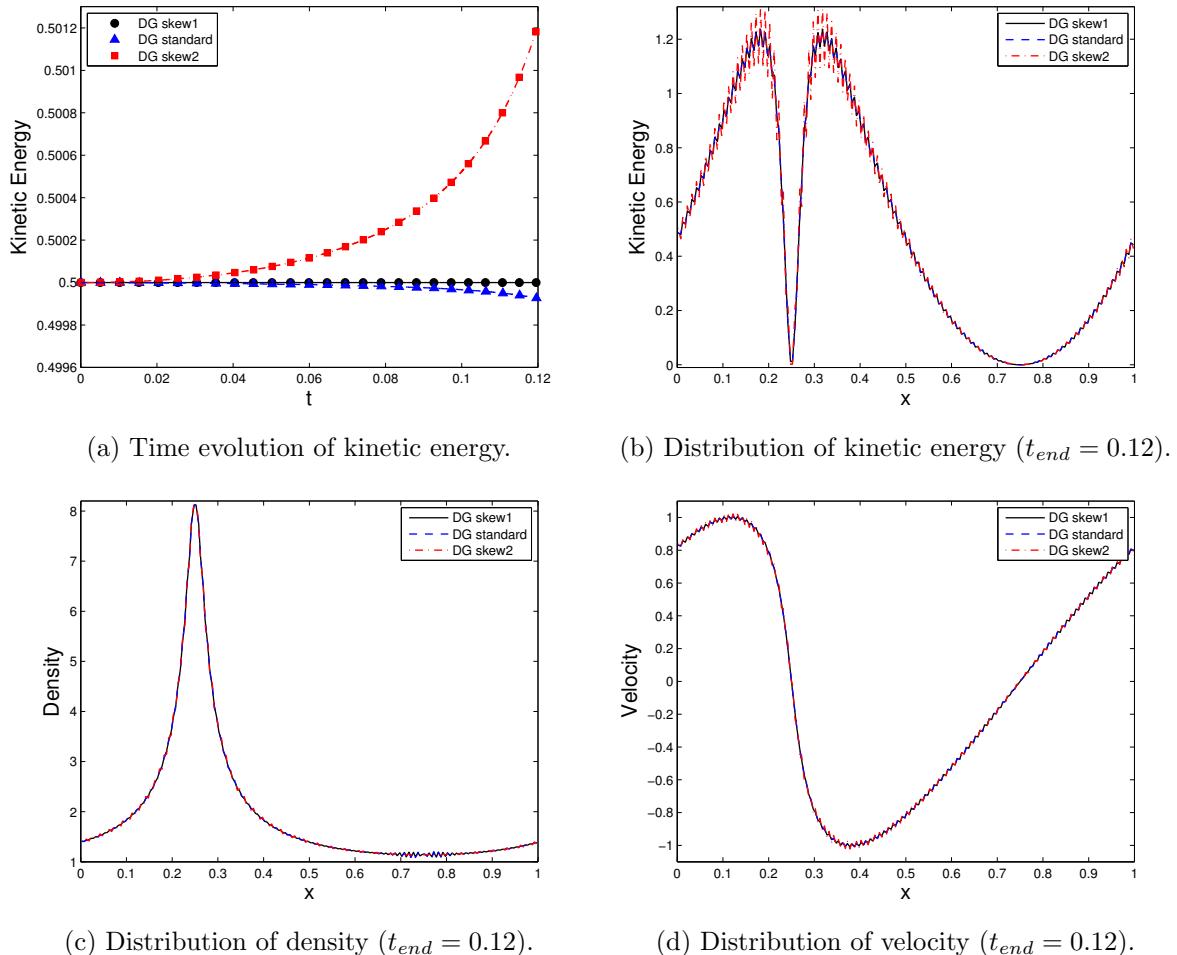


Figure 1.2: Numerical results of the DG scheme for a polynomial degree of  $N = 1$  on a computational grid of 100 cells using different variants of the skew-symmetric terms.

### Non-linear acoustic pressure wave

In [61], Gassner proposed a test case which is sensitive to dissipation and dispersion errors to study the performance of the kinetic energy preserving DG scheme based on Legendre-Gauss-Lobatto nodes on coarse grids and for a low polynomial degree of  $N = 1$ . In comparison to the standard DG scheme discretizing the conservative Euler equations, the approximation by the skew-symmetric scheme using the KEP flux appeared to be closer to the reference solution with respect to the pressure wave and the kinetic energy distribution.

The same test case is used in this work to investigate the performance of the KEP-DG scheme based on the classical Legendre-Gauss points. Therefore, the Euler equations are augmented by viscous terms of the compressible Navier-Stokes equations. The equations to be solved are given by

$$\frac{\partial}{\partial t} \mathbf{U}(x, t) + \frac{\partial}{\partial x} \mathbf{F}(\mathbf{U}(x, t)) = \frac{\partial}{\partial x} \mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x), \quad (1.145)$$

where  $\mathbf{F}$  again denotes the inviscid convective fluxes contained in the Euler equations and  $\mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x)$  contains the viscous fluxes given by

$$\mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x) = \begin{pmatrix} 0 \\ \mu \frac{4}{3} v_x \\ \mu \frac{4}{3} v v_x + k T_x \end{pmatrix}. \quad (1.146)$$

Herein, the viscosity coefficient  $\mu = \mu(T)$  may depend on the temperature  $T = \frac{p}{\rho R} = \frac{p\gamma}{\rho(\gamma-1)c_p} = \frac{\gamma}{c_p} e$ , where  $R$  denotes the gas constant of the ideal gas law and  $c_p$  is the specific heat at constant pressure. The heat conduction coefficient is furthermore given by  $k = \frac{c_p \mu}{Pr}$ , with the Prandtl number  $Pr$ . As in [61], the dependence of the viscosity  $\mu$  on the temperature is neglected to simplify the equation and its discretization.

We then note that the viscous terms can be re-written as  $\mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x) = \mathbf{A}(\mathbf{U}) \mathbf{U}_x$  using the diffusion matrix

$$\mathbf{A}(\mathbf{U}) = \frac{\mu}{\rho} \begin{pmatrix} 0 & 0 & 0 \\ -\frac{4}{3}v & \frac{4}{3} & 0 \\ -\left(\frac{4}{3}v^2 + \frac{\gamma}{Pr}(e - v^2)\right) & \left(\frac{4}{3} - \frac{\gamma}{Pr}\right)v & \frac{\gamma}{Pr} \end{pmatrix}.$$

Now, to specify the set-up of the numerical experiment analogously to [61], the initial conditions for the acoustic pressure wave are given by the following initial density, velocity and pressure distribution

$$\rho(x, 0) = 1, \quad v(x, 0) = 1, \quad p(x, 0) = 1 + 0.1 \sin(2\pi x)$$

on the computational domain  $\Omega = [0, 1]$  with periodic boundary conditions. The viscosity coefficient is set to  $\mu = 0.002$  and the Prandtl number is  $Pr = 0.72$ . The viscous terms are discretized by the BR1 approach developed by Bassi and Rebay, see [13] or the later discussion in Section 2.1. In order to study long time integration, the numerical computations are then carried out until the final time  $t_{end} = 20$  is reached. For this test case computed by the Legendre-Gauss DG scheme, the results showed no difference in accuracy for the DG scheme

with or without skew-symmetric terms - even in case of the inconsistent boundary treatment. However, we may study the effect of a higher order quadrature rule when using Legendre-Gauss nodes instead of the Legendre-Gauss-Lobatto variant considered in [61]. Thus, Figure 1.3 reports the output of the KEP-DG scheme for  $N = 1$  in case of Legendre-Gauss (LG) as well as Legendre-Gauss-Lobatto (LGL) nodes using the KEP-VL flux (1.137) and correct combinations of skew-symmetric and boundary terms, i.e. terms (1.142) for the Legendre-Gauss case and terms (1.143) for the Legendre-Gauss-Lobatto case.

In order to account for the differences in arithmetic operations in case of Legendre-Gauss or Legendre-Gauss-Lobatto nodes, respectively, the DG scheme on Legendre-Gauss nodes uses 40 cells while for the Legendre-Gauss-Lobatto variant 80 cells are taken. Thus, the stability constraint for explicit time integration is roughly the same, see [65]. The additional cost introduced by the finer grid for the Legendre-Gauss-Lobatto nodes as well as the additional boundary interpolation and boundary correction terms for Legendre-Gauss nodes can then be quantified and compared. More precisely, the skew-symmetric DG scheme (1.128) on Legendre-Gauss nodes needs to interpolate the conservative variables as well as velocity to the two boundaries of grid cells, which results in  $16(N + 1)$  arithmetic operations per coarse grid cell, and to evaluate the additional surface correction, i.e. the last term on the right-hand side of (1.128), second line, for which we have  $9(N + 1)$  arithmetic operations per coarse grid cell. On the other hand, the Legendre-Gauss-Lobatto variant needs twice as many grid cells, hence twice as many flux evaluations and multiplications by  $\mathbf{S}^T$ , resulting in  $7(N + 1) + 6(N + 1)^2$  arithmetic operations per coarse grid cell. Furthermore, the double number of skew-symmetric term evaluations is necessary, i.e. additional  $4((N + 1)^2 + N + 1)$  arithmetic operations in the weak formulation and twice as many evaluations of numerical fluxes, resulting in 18 additional arithmetic operations per coarse grid interface for the KEP flux. For  $N = 1$  and periodic boundary conditions in one space dimension, i.e. an equal number of grid cells and interfaces, this results in 50 additional arithmetic operations per coarse grid cell for the KEP-DG scheme on Legendre-Gauss nodes in comparison to the Legendre-Gauss-Lobatto variant on the coarser grid and 80 additional arithmetic operations for the Legendre-Gauss-Lobatto variant on the finer grid in comparison to the coarser. Thus, the Legendre-Gauss-Lobatto set-up is designed to be more expensive for this problem. However, in a comparison with a reference solution obtained by the standard DG scheme for a polynomial degree  $N = 3$  and 500 cells, this Legendre-Gauss-Lobatto variant clearly is not as accurate as the Legendre-Gauss variant on the coarser grid as shown in Fig. 1.3 where the DG solution with Legendre-Gauss nodes almost cannot be distinguished from the reference solution. For this test case, it has hence payed off to consider Legendre-Gauss nodes, though kinetic energy preservation seems to be less critical for this test case. Again one should remark that preserving a qualitative behavior does not guarantee better accuracy in general.

### Viscous Sod shock tube

In order to investigate the behavior of KEP schemes near their limits of applicability, i.e. exact solutions with shocks, Allaneau and Jameson studied their performance for viscous shock test cases e.g. in [4]. For finite volume approximations on coarse meshes, a comparison of numerical fluxes gave better, oscillation-free, results in case of diffusive numerical fluxes. However, the KEP flux (1.134) lead to stable computations whereas the regular central scheme

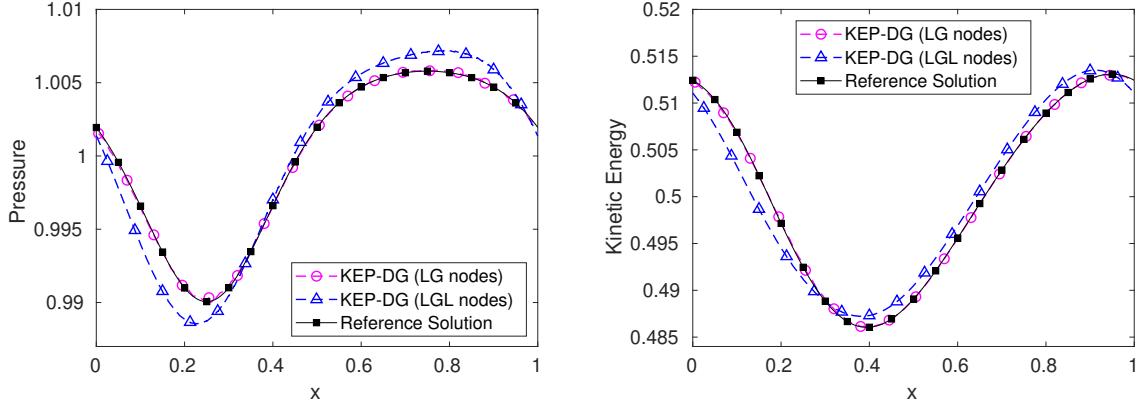


Figure 1.3: KEP-DG approximate solutions for  $N = 1$ , Legendre-Gauss (40 cells) vs. Legendre-Gauss-Lobatto nodes (80 cells) using the kinetic energy preserving flux  $f_C^*$ . Left: pressure. Right: kinetic energy.

blew up. In [4], high order DG schemes on coarse meshes were considered as well, where again the KEP flux performed better than a regular central scheme in damping oscillations due to odd/even decoupling. With these former investigations in mind, it should be interesting to study DG schemes incorporating different skew-symmetric terms in addition to a kinetic energy preserving numerical flux. Hence, we now consider a corresponding test case of a viscous Sod shock tube. The system of equations to be solved is again the Navier-Stokes equations as in the previous test case using a constant viscosity coefficient  $\mu = 0.0001$ . The initial conditions on the computational domain  $\Omega = [0, 1]$  are given by

$$\mathbf{U}(x, 0) = \begin{cases} \mathbf{U}_{left} & \text{for, } x < 0.5, \\ \mathbf{U}_{right}, & \text{otherwise,} \end{cases}$$

with left and right states  $\mathbf{U}_{left}$  and  $\mathbf{U}_{right}$  given by

$$\begin{pmatrix} \rho_{left} \\ v_{left} \\ p_{left} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} \rho_{right} \\ v_{right} \\ p_{right} \end{pmatrix} = \begin{pmatrix} 0.125 \\ 0 \\ 0.1 \end{pmatrix}.$$

Both of the computational boundaries are supplemented by inflow boundary conditions and the computations are carried out until the final time  $t_{end} = 0.15$ .

Figure 1.4 shows the DG solution for  $N = 1$  on a coarse grid of 100 cells using the different skew-symmetric terms (1.142) and (1.143) as well as the kinetic energy preserving KEP-VL flux (1.137). We clearly see oscillations at the shock position produced by the DG scheme using skew symmetric terms with inconsistent boundary treatment. No such instability phenomenon is present in the case of the usual van Leer flux, see Figure 1.5a, whereas Figure 1.5b again shows larger oscillations for the inconsistent boundary treatment when choosing the KEP flux (1.134). For the DG scheme with  $N = 3$  and 100 cells, the resolution is high enough and the precise choice of numerical flux function has less effect as shown in Figure 1.6.

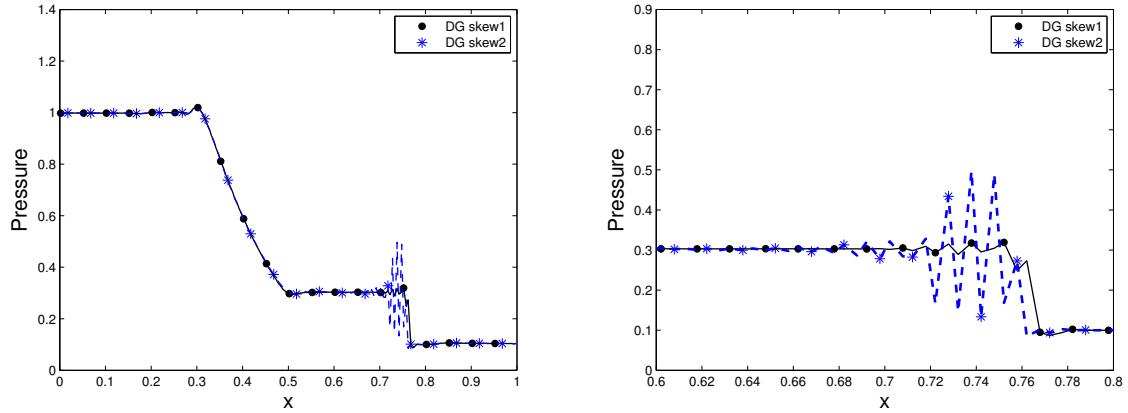


Figure 1.4: DG scheme for  $N = 1$  and 100 cells using different skew-symmetric terms and kinetic energy preserving flux  $f_C^*$ . Right: close-up at instability region.

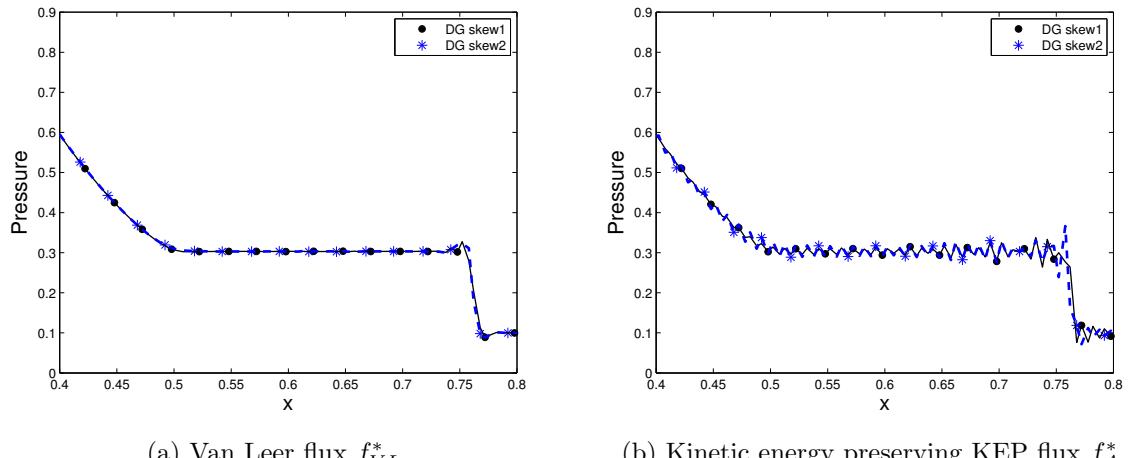


Figure 1.5: Numerical results given by the DG scheme for a polynomial degree of  $N = 1$  on 100 cells using different skew-symmetric terms for two choices of the numerical flux function.

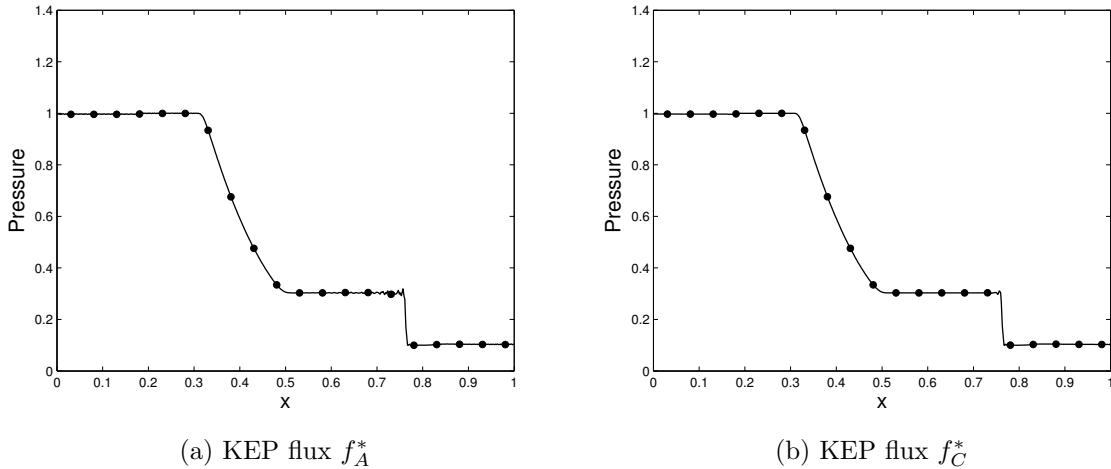


Figure 1.6: Numerical results given by the DG scheme for a polynomial degree of  $N = 3$  on 100 cells using correct skew-symmetric terms and two different choices of numerical flux functions both having the KEP property.

### 1.4.3 Extension to cartesian grids in two space dimensions

The skew-symmetric, kinetic energy preserving DG scheme on Legendre-Gauss nodes easily extends to two-dimensional cartesian grids using tensor-product basis functions  $L_i(\xi)L_j(\eta)$  on the reference element  $K = [-1, 1]^2$ . In this case, a nodal DG scheme with an SBP property may be constructed as in Section 1.2.4 based on the 1D formulation by using Kronecker products. Applied to a skew-symmetric formulation of the Euler equations in two space dimensions, this tensor-product DG formulation allows for a kinetic energy preserving DG scheme in 2D with similar additional terms as in the 1D case. In particular, the necessary boundary correction terms for the interior node distributions are a direct extension of the one-dimensional case. In this section, we will derive this 2D extension. Furthermore, in Section 1.4.4, the advantage of using a kinetic energy preserving scheme is demonstrated in the context of two-dimensional turbulent flow.

For the construction of a kinetic energy preserving DG scheme on two-dimensional cartesian grids, we consider the skew-symmetric form of the Euler equations in two space dimensions given by

$$\begin{aligned}\frac{\partial}{\partial t}\rho &= -\frac{\partial}{\partial x}(\rho v_1) - \frac{\partial}{\partial y}(\rho v_2), \\ \frac{1}{2} \left[ \frac{\partial}{\partial t}(\rho v_1) + \rho \frac{\partial v_1}{\partial t} \right] &= -\frac{1}{2} \left[ \frac{\partial}{\partial x}(\rho v_1^2) + \rho v_1 \frac{\partial v_1}{\partial x} \right] - \frac{1}{2} \left[ \frac{\partial}{\partial y}(\rho v_1 v_2) + \rho v_2 \frac{\partial v_1}{\partial y} \right] - \frac{\partial p}{\partial x}, \\ \frac{1}{2} \left[ \frac{\partial}{\partial t}(\rho v_2) + \rho \frac{\partial v_2}{\partial t} \right] &= -\frac{1}{2} \left[ \frac{\partial}{\partial x}(\rho v_1 v_2) + \rho v_1 \frac{\partial v_2}{\partial x} \right] - \frac{1}{2} \left[ \frac{\partial}{\partial y}(\rho v_2^2) + \rho v_2 \frac{\partial v_2}{\partial y} \right] - \frac{\partial p}{\partial y}, \\ \frac{\partial}{\partial t}(\rho E) &= -\frac{\partial}{\partial x}(\rho v_1 E + v_1 p) - \frac{\partial}{\partial y}(\rho v_2 E + v_2 p).\end{aligned}$$

The scheme (1.73) applied to the continuity equation is given by

$$\frac{\Delta x \Delta y}{4} \frac{d}{dt} \mathbf{u}_1 + \mathbf{D}_\xi \mathbf{f}_1^\xi + \mathbf{D}_\eta \mathbf{f}_1^\eta = \mathbf{M}^{-1} \left\langle \left( n_\xi f_{1,h}^\xi + n_\eta f_{1,h}^\eta - f_1^* \right) \mathbf{L}(\xi, \eta) \right\rangle_{\partial K}$$

In the same manner, we may discretize the energy equation as it is given in divergence form. For the skew-symmetric forms of the momentum equations, discretization of  $\frac{\partial}{\partial x}$  by  $\mathbf{D}_\xi$  and  $\frac{\partial}{\partial y}$  by  $\mathbf{D}_\eta$  as well as interface terms corresponding to the right-hand side of (1.73) yield

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \frac{1}{2} \left[ \frac{d\mathbf{u}_2}{dt} + \underline{\underline{\mathbf{u}}}_1 \frac{d\mathbf{v}_1}{dt} \right] + \frac{1}{2} \left[ \mathbf{D}_\xi \underline{\underline{\mathbf{p}}} \mathbf{v}_1^2 + \underline{\underline{\mathbf{u}}}_2 \mathbf{D}_\xi \mathbf{v}_1 + \mathbf{D}_\eta \underline{\underline{\mathbf{u}}}_3 \mathbf{v}_1 + \underline{\underline{\mathbf{u}}}_3 \mathbf{D}_\eta \mathbf{v}_1 \right] + \mathbf{D}_\xi \mathbf{p} \\ = \mathbf{M}^{-1} \left\langle (n_\xi g_{2,h}^\xi + n_\eta g_{2,h}^\eta - g_2^*) \mathbf{L} \right\rangle_{\partial K}, \end{aligned} \quad (1.147)$$

and

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \frac{1}{2} \left[ \frac{d\mathbf{u}_3}{dt} + \underline{\underline{\mathbf{u}}}_1 \frac{d\mathbf{v}_2}{dt} \right] + \frac{1}{2} \left[ \mathbf{D}_\xi \underline{\underline{\mathbf{u}}}_2 \mathbf{v}_2 + \underline{\underline{\mathbf{u}}}_2 \mathbf{D}_\xi \mathbf{v}_2 + \mathbf{D}_\eta \underline{\underline{\mathbf{p}}} \mathbf{v}_2^2 + \underline{\underline{\mathbf{u}}}_3 \mathbf{D}_\eta \mathbf{v}_2 \right] + \mathbf{D}_\eta \mathbf{p} \\ = \mathbf{M}^{-1} \left\langle (n_\xi g_{3,h}^\xi + n_\eta g_{3,h}^\eta - g_3^*) \mathbf{L} \right\rangle_{\partial K}, \end{aligned} \quad (1.148)$$

with terms  $g_{2,h}, g_2^*, g_{3,h}, g_3^*$  to be specified. As in the 1D case, multiplication of the semi-discrete continuity equation from left by  $\frac{1}{2}\underline{\underline{\mathbf{v}}}_1$ , adding the result to (1.147) and using continuity in time, we have

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \frac{d\mathbf{u}_2}{dt} + \mathbf{D}_\xi \mathbf{f}_2^\xi + \mathbf{D}_\eta \mathbf{f}_2^\eta + \mathbf{s}_2^\xi + \mathbf{s}_2^\eta \\ = \mathbf{M}^{-1} \left( \left\langle (n_\xi g_{2,h}^\xi + n_\eta g_{2,h}^\eta - g_2^*) \mathbf{L} \right\rangle_{\partial K} + \frac{1}{2} \underline{\underline{\mathbf{v}}}_1 \left\langle (n_\xi f_{1,h}^\xi + n_\eta f_{1,h}^\eta - f_1^*) \mathbf{L} \right\rangle_{\partial K} \right), \end{aligned}$$

with skew-symmetric terms  $\mathbf{s}_2^\xi, \mathbf{s}_2^\eta$  given by

$$\begin{aligned} \mathbf{s}_2^\xi &= \frac{1}{2} \left[ -\mathbf{D}_\xi \underline{\underline{\mathbf{u}}}_2 \mathbf{v}_1 + \underline{\underline{\mathbf{u}}}_2 \mathbf{D}_\xi \mathbf{v}_1 + \underline{\underline{\mathbf{v}}}_1 \mathbf{D}_\xi \mathbf{u}_2 \right], \\ \mathbf{s}_2^\eta &= \frac{1}{2} \left[ -\mathbf{D}_\eta \underline{\underline{\mathbf{u}}}_3 \mathbf{v}_1 + \underline{\underline{\mathbf{u}}}_3 \mathbf{D}_\eta \mathbf{v}_1 + \underline{\underline{\mathbf{v}}}_1 \mathbf{D}_\eta \mathbf{u}_3 \right]. \end{aligned}$$

Using the SBP properties of  $\mathbf{D}_\xi$  and  $\mathbf{D}_\eta$  a corresponding weak formulation may be obtained. Analogous to the 1D case, we multiply by the diagonal matrix  $\mathbf{M}$  to obtain

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \mathbf{M} \frac{d\mathbf{u}_2}{dt} + (\mathbf{B}_\xi - \mathbf{S}_\xi^T) \left( \mathbf{f}_2^\xi - \frac{1}{2} \underline{\underline{\mathbf{u}}}_2 \mathbf{v}_1 \right) + (\mathbf{B}_\eta - \mathbf{S}_\eta^T) \left( \mathbf{f}_2^\eta - \frac{1}{2} \underline{\underline{\mathbf{u}}}_3 \mathbf{v}_1 \right) \\ + \frac{1}{2} \left[ \underline{\underline{\mathbf{u}}}_2 \mathbf{S}_\xi \mathbf{v}_1 + \underline{\underline{\mathbf{v}}}_1 (\mathbf{B}_\xi - \mathbf{S}_\xi^T) \mathbf{u}_2 + \underline{\underline{\mathbf{u}}}_3 \mathbf{S}_\eta \mathbf{v}_1 + \underline{\underline{\mathbf{v}}}_1 (\mathbf{B}_\eta - \mathbf{S}_\eta^T) \mathbf{u}_3 \right] \\ = \left( \left\langle (n_\xi g_{2,h}^\xi + n_\eta g_{2,h}^\eta - g_2^*) \mathbf{L} \right\rangle_{\partial K} + \frac{1}{2} \underline{\underline{\mathbf{v}}}_1 \left\langle (n_\xi f_{1,h}^\xi + n_\eta f_{1,h}^\eta - f_1^*) \mathbf{L} \right\rangle_{\partial K} \right). \end{aligned} \quad (1.149)$$

Choosing  $g_{2,h}^\xi = (\frac{1}{2}\rho v_1^2 + p)_h$  and  $g_{2,h}^\eta = (\frac{1}{2}\rho v_1 v_2)_h$  in accordance with the 1D case and using Lemma 1.13 to cancel out boundary terms yields

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \mathbf{M} \frac{d\mathbf{u}_2}{dt} &= \mathbf{S}_\xi^T \left( \mathbf{f}_2^\xi - \frac{1}{2} \underline{\underline{\mathbf{u}}}_2 \mathbf{v}_1 \right) + \mathbf{S}_\eta^T \left( \mathbf{f}_2^\eta - \frac{1}{2} \underline{\underline{\mathbf{u}}}_3 \mathbf{v}_1 \right) \\ &\quad - \frac{1}{2} \left[ \underline{\underline{\mathbf{u}}}_2 \mathbf{S}_\xi \mathbf{v}_1 - \underline{\underline{\mathbf{v}}}_1 \mathbf{S}_\xi^T \mathbf{u}_2 + \underline{\underline{\mathbf{u}}}_3 \mathbf{S}_\eta \mathbf{v}_1 - \underline{\underline{\mathbf{v}}}_1 \mathbf{S}_\eta^T \mathbf{u}_3 \right] \\ &\quad - \left( \langle g_2^* \mathbf{L} \rangle_{\partial K} + \frac{1}{2} \underline{\underline{\mathbf{v}}}_1 \langle f_1^* \mathbf{L} \rangle_{\partial K} \right). \end{aligned} \quad (1.150)$$

Multiplying (1.150) from left with  $\mathbf{1}^T$  cancels the volume terms and we obtain

$$\frac{\Delta x \Delta y}{4} \mathbf{1}^T \mathbf{M} \frac{d\mathbf{u}_2}{dt} = - \left( \langle g_2^* \mathbf{1}^T \mathbf{L} \rangle_{\partial K} + \frac{1}{2} \mathbf{v}_1^T \langle f_1^* \mathbf{L} \rangle_{\partial K} \right) = - \left\langle \left( g_2^* + \frac{1}{2} v_{1,h} f_1^* \right) \mathbf{L} \right\rangle_{\partial K}.$$

Hence consistency demands  $g_2^* = f_2^* - v_{1,h} f_1^*$  in accordance with the one-dimensional case. Analogous derivations for the  $y$ -direction of the momentum equations yield the corresponding weak formulation

$$\begin{aligned} \frac{\Delta x \Delta y}{4} \mathbf{M} \frac{d\mathbf{u}_3}{dt} &= \mathbf{S}_\xi^T \left( \mathbf{f}_3^\xi - \frac{1}{2} \underline{\underline{\mathbf{u}}}_2 \mathbf{v}_2 \right) + \mathbf{S}_\eta^T \left( \mathbf{f}_3^\eta - \frac{1}{2} \underline{\underline{\mathbf{u}}}_3 \mathbf{v}_2 \right) \\ &\quad - \frac{1}{2} \left[ \underline{\underline{\mathbf{u}}}_2 \mathbf{S}_\xi \mathbf{v}_2 - \underline{\underline{\mathbf{v}}}_2 \mathbf{S}_\xi^T \mathbf{u}_2 + \underline{\underline{\mathbf{u}}}_3 \mathbf{S}_\eta \mathbf{v}_2 - \underline{\underline{\mathbf{v}}}_2 \mathbf{S}_\eta^T \mathbf{u}_3 \right] \\ &\quad - \left\langle \left( f_3^* \mathbf{I} + \frac{1}{2} \left( \underline{\underline{\mathbf{v}}}_2 - v_{2,h} \mathbf{I} \right) f_1^* \right) \mathbf{L} \right\rangle_{\partial K}. \end{aligned} \quad (1.151)$$

Furthermore, the same derivations as in the 1D case yield a semi-discrete balance equation for the kinetic energy  $\mathbf{e}_{kin} = \frac{1}{2} \underline{\underline{\rho}} (\mathbf{v}_1^2 + \mathbf{v}_2^2)$ . Similar to equation (1.130) we arrive at

$$\frac{d}{dt} \mathbf{1}^T \mathbf{M} \mathbf{e}_{kin} = \left( (\mathbf{D}_\xi \mathbf{v}_1)^T + (\mathbf{D}_\eta \mathbf{v}_2)^T \right) \mathbf{M} \mathbf{p} - \mathbf{v}_1^T \langle g_2^* \mathbf{L} \rangle_{\partial K} - \mathbf{v}_2^T \langle g_3^* \mathbf{L} \rangle_{\partial K}.$$

Hence, for a correct kinetic energy balance the left and right-hand sided transport terms within the surface fluxes have to cancel out as in the 1D case. Decomposing  $g_2^{*,\pm} = \tilde{g}_2^{*,\pm} + n_\xi p^*$  and  $g_3^{*,\pm} = \tilde{g}_3^{*,\pm} + n_\eta p^*$  with  $\tilde{g}_k^{*,\pm} = \tilde{f}_k^* - \frac{1}{2} v_{k-1}^\pm f_1^*$  a condition of the form

$$v_1^- \tilde{g}_2^{*,-} + v_2^- \tilde{g}_3^{*,-} = v_1^+ \tilde{g}_2^{*,+} + v_2^+ \tilde{g}_3^{*,+}$$

needs to be fulfilled. Thus, a suitable choice for kinetic energy preservation similar to the 1D case is

$$\tilde{f}_k^* = \bar{v}_{k-1} f_1^*, \quad k = 2, 3.$$

In particular, this holds for the KEP flux using rotational invariance of the Euler equations. This numerical flux is given by

$$\begin{pmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \end{pmatrix} = \begin{pmatrix} f_1^{*,1D}(u^-, u^+, n) \\ n_\xi f_2^{*,1D}(u^-, u^+, n) - n_\eta f_3^{*,1D}(u^-, u^+, n) \\ n_\eta f_2^{*,1D}(u^-, u^+, n) + n_\xi f_3^{*,1D}(u^-, u^+, n) \\ f_4^{*,1D}(u^-, u^+, n) \end{pmatrix},$$

where  $\mathbf{f}^{*,1D}$  denotes the 1D KEP flux in normal direction

$$\begin{aligned} f_{1D,1}^* &= \bar{\rho}\bar{v}_n, \\ f_{1D,2}^* &= \bar{\rho}\bar{v}_n^2 + \bar{p}, \\ f_{1D,3}^* &= \bar{\rho}\bar{v}_n\bar{v}_t, \\ f_{1D,4}^* &= \bar{\rho}\bar{v}_n\bar{H}, \end{aligned}$$

using the normal and tangential velocity  $v_n = n_\xi v_1 + n_\eta v_2$  and  $v_t = n_\xi v_2 - n_\eta v_1$ .

#### 1.4.4 Numerical simulation of 2D homogeneous turbulence

Two-dimensional homogeneous turbulence is an energy-decaying system which is extensively used to study accuracy and efficiency of numerical methods, e.g. in [88, 170, 215]. Our purpose is to demonstrate the improved resolution of the proposed kinetic energy preserving skew-symmetric DG scheme for this test case compared to the standard DG scheme. The computational domain is the square  $[0, 2\pi]^2$  supplied with periodic boundary conditions. The initial energy spectrum is given in Fourier space by

$$E(k) = \frac{a_s}{2} \frac{1}{k_p} \left( \frac{k}{k_p} \right)^{2s+1} \exp \left[ - \left( s + \frac{1}{2} \right) \left( \frac{k}{k_p} \right)^2 \right],$$

where  $k = \sqrt{k_x^2 + k_y^2}$ . The initial energy spectrum attains its maximum at the wavenumber  $k_p$ . As in the references, the parameters are set to  $k_p = 12$ ,  $a_s = \frac{(2s+1)^{s+1}}{2^s s!}$ ,  $s = 3$ . From this initial energy spectrum an initial velocity distribution is obtained using transfer procedures described in [88, 170] where a random phase is introduced into the vorticity field. The initial velocity distribution in physical space is then given by the inverse Fourier transform. For the compressible flow computations the initial density is set to  $\rho_0 = 1$  while the pressure is computed setting the initial Mach number to  $\text{Ma} = 0.1$ . The viscosity coefficient  $\mu$  may be varied to study the quality of the numerical solutions for different Reynolds numbers. We then compute the numerical solution to this test problem both with the standard DG scheme and the kinetic energy preserving skew-symmetric DG scheme based on the Legendre-Gauss nodes until time  $T = 10$ . The resulting energy spectrum  $E(k)$  at this final time is then computed from the velocity distribution by the same procedures as in [170]. Figure 1.7 depicts the comparison of the standard DG scheme on Legendre-Gauss nodes for  $N = 1$  and its kinetic energy preserving variant KEP-DG in terms of their energy spectrum for a lower Reynolds number of  $\text{Re} = 100$ . A reference solution is obtained by the standard DG scheme of 5th order on 80 grid cells. A more accurate representation of the energy spectrum is obvious for the second order KEP-DG scheme, both on the very coarse grid with 40 cells and on the finer one of 80 cells. Figure 1.8 shows the corresponding comparison of the second order standard DG scheme vs. the KEP-DG scheme on Legendre-Gauss points for  $\text{Re} = 600$ . Now, the reference solution is obtained by the standard 5th order DG scheme on 160 grid cells, although this numerical solution is indistinguishable from the one on 80 grid cells in this case. Also for the higher Reynolds number, the energy spectrum is represented more accurately by the KEP-DG scheme.

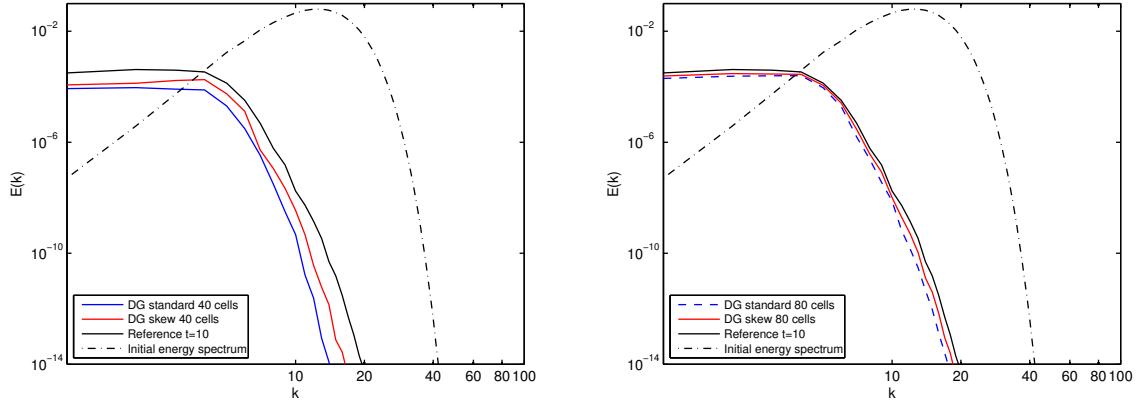


Figure 1.7: Comparison of DG scheme and KEP-DG on Legendre-Gauss nodes for  $N = 1$  and  $\text{Re} = 100$ . Energy spectrum at time  $T = 10$ . Left: 40 cells. Right: 80 cells.

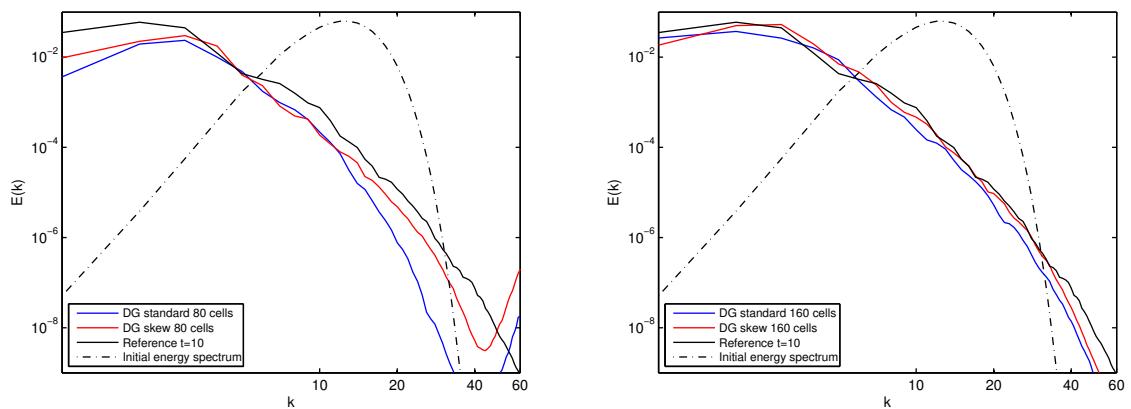


Figure 1.8: Comparison of DG scheme and KEP-DG on Legendre-Gauss nodes for  $N = 1$  and  $\text{Re} = 600$ . Energy spectrum at time  $T = 10$ . Left: 80 cells. Right: 160 cells.

### 1.4.5 Extension to moving grids: application to fluid-structure interaction

The interaction between a moving piston attached to a spring and an inviscid fluid contained in the piston chamber is a classical test case in the context of fluid-structure interaction which is described for instance in [112, 19]. The coupled piston problem and its one-dimensional set-up are illustrated in Figure 1.9, where  $L_0$  denotes the chamber length at rest and  $q(t)$  is the piston position at time  $t$ . Since the computational domain  $\Omega(t) = [-L_0, q(t)]$  is time dependent, the position of the equidistant grid nodes  $x_1 = x_1(t), \dots, x_{K+1} = x_{K+1}(t)$  of the spatial discretization also varies in time.

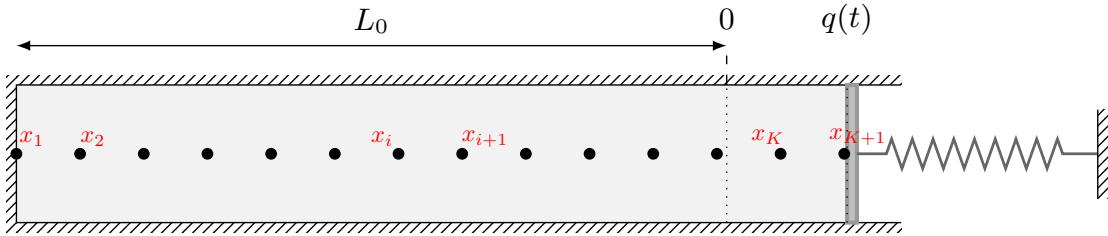


Figure 1.9: One-dimensional piston problem.

In the mathematical formulation of the classical piston problem, the fluid is described by the one-dimensional inviscid Euler equations while the displacement of the piston is modeled by an undamped harmonic oscillator.

On the moving grid, the Euler equations are now given in arbitrary Lagrangian-Eulerian (ALE) formulation according to the introductory paper by Lefrançois and Boufflet [112]. Using a reference coordinate  $\xi = \xi(x, t) \in [0, L]$  on the fixed domain given by the state at rest, this formulation of the Euler equations in one space dimension is

$$\frac{\partial}{\partial t} (J\mathbf{U}) + J \frac{\partial}{\partial x} \hat{\mathbf{F}}(\mathbf{U}, w) = 0, \quad (1.152)$$

where  $J = J(x, t)$  denotes the Jacobian of the grid motion and  $w = w(x, t)$  refers to the local domain velocity specified by

$$J = \frac{\partial x(\xi, t)}{\partial \xi}, \quad w = \frac{\partial x(\xi, t)}{\partial t}.$$

In the ALE formulation (1.152), the vector of conserved variables  $\mathbf{U} = (\rho, \rho v, \rho E)^T$  is unaltered in comparison to the formulation (1.144) on fixed grids whereas the adjusted flux vector  $\hat{\mathbf{F}}$  defined by

$$\hat{\mathbf{F}}(\mathbf{U}, w) = \begin{pmatrix} \rho(v - w) \\ \rho v(v - w) + p \\ \rho E(v - w) + vp \end{pmatrix}$$

now takes into account the relative velocity  $\hat{v} = v - w$  due to grid motion.

The piston displacement is modeled by a mass-spring system driven by the difference of the ambient pressure  $p_A$  to the pressure  $p_I$  inside the chamber at the interface. Denoting the

piston displacement by  $q(t)$ , its behavior is therefore described by the second-order linear ODE

$$m\ddot{q} + kq = A(p_I - p_A), \quad (1.153)$$

with mass  $m$ , stiffness  $k$  and cross-sectional area  $A$ , and the initial conditions

$$q(0) = q_0, \quad q'(0) = r_0.$$

Coupling conditions for this fluid-structure interaction problem are specified by enforcing

$$\begin{aligned} x(\xi = L, t) &= q(t), \\ x(\xi = 0, t) &= -L_0, \end{aligned}$$

for the moving right chamber boundary and the fixed left-hand side of the chamber, respectively. For the coupling between the local domain velocity  $w$  and the fluid velocity  $v$  at the chamber boundaries, we therefore demand

$$\begin{aligned} w(q(t), t) &= \dot{q}(t) = v(q(t), t), \\ w(-L_0, t) &= 0 = v(-L_0, t). \end{aligned}$$

Concerning the grid movement, the most natural choice is to set

$$x(\xi, t) = \frac{\xi + L_0}{L_0} (q(t) + L_0) - L_0, \quad (1.154)$$

whereby we obtain a space invariant grid Jacobian

$$J(x, t) = \frac{\partial x(\xi, t)}{\partial \xi} = \frac{q(t) + L_0}{L_0}, \quad (1.155)$$

which facilitates the set-up of the DG scheme.

In the following, we again employ a DG space discretization for the fluid equations in standard and in skew-symmetric form, where the grid nodes in Figure 1.9 are the element boundaries of the time-dependent DG cells  $I_i(t) = [x_i(t), x_{i+1}(t)]$ . For a fixed grid, this notation is consistent with the one introduced in Section 1.2.1. Now, the reference cell  $[-1, 1]$  is mapped to the cells  $I_i(t)$  by the transformation

$$\Lambda_i(\tilde{\xi}, t) = \tilde{\xi} \frac{x_{i+1}(t) - x_i(t)}{2} + \frac{x_i(t) + x_{i+1}(t)}{2}, \quad \tilde{\xi} \in [-1, 1], \quad (1.156)$$

analogous to the definition in (1.43).

Reusing the notation for the mass matrix  $\mathbf{M}$  and the first-derivative operator  $\mathbf{D}$  introduced in Section 1.2.1, the DG space discretization of the compressible Euler equations on moving domains in ALE formulation is now given by

$$\frac{\Delta x}{2} \frac{d}{dt} (J \mathbf{u}_k) + \mathbf{D} \hat{\mathbf{f}}_k + \hat{\mathbf{s}}_k = \mathbf{M}^{-1} \left( [(\hat{f}_{k,h} - \hat{f}_k^*) \mathbf{L}]_{-1}^1 + \hat{\mathbf{s}}_k^{bc} \right), \quad k = 1, 2, 3,$$

with modified nodal flux evaluations  $\hat{\mathbf{f}}_k$ , interpolated fluxes  $\hat{f}_{k,h}$  and numerical fluxes  $\hat{f}_k^*$ , taking into account the relative fluid velocity, as well as correspondingly adjusted skew-symmetric volume terms  $\hat{\mathbf{s}}_k$  and boundary corrections  $\hat{\mathbf{s}}_k^{bc}$ . More precisely, the nodal flux evaluations are given by

$$\mathbf{f}_1 = \underline{\rho} \hat{\mathbf{v}}, \quad \mathbf{f}_2 = \underline{\rho} \underline{\mathbf{v}} \hat{\mathbf{v}} + \mathbf{p}, \quad \mathbf{f}_3 = \underline{\rho} \underline{\mathbf{E}} \hat{\mathbf{v}} + \underline{\mathbf{v}} \mathbf{p},$$

incorporating the relative nodal velocity  $\hat{\mathbf{v}} = \mathbf{v} - \mathbf{w}$ , while the interpolated fluxes  $\hat{f}_{k,h}$  and numerical fluxes  $\hat{f}_k^*$  are modified accordingly. Furthermore, for the standard DG scheme, we set

$$\hat{\mathbf{s}}_k = \hat{\mathbf{s}}_k^{bc} = \mathbf{0}, \quad k = 1, 2, 3,$$

while for the skew-symmetric variant, only for the momentum equation, the skew-symmetric volume term and boundary correction are nonzero, setting

$$\begin{aligned} \hat{\mathbf{s}}_2 &= \frac{1}{2} \left[ -\mathbf{D} \underline{\rho} \underline{\mathbf{v}} \hat{\mathbf{v}} + \underline{\rho} \hat{\mathbf{v}} \mathbf{D} \mathbf{v} + \underline{\mathbf{v}} \mathbf{D} \underline{\rho} \hat{\mathbf{v}} \right], \\ \hat{\mathbf{s}}_2^{bc} &= \frac{1}{2} \left( \underline{\mathbf{v}} [(\hat{f}_{1,h} - \hat{f}_1^*) \mathbf{L}]_{-1}^1 - [(\rho v \hat{v})_h - v^\pm \hat{f}_1^*] \mathbf{L}]_{-1}^1 \right). \end{aligned}$$

For time discretization of the coupled system, we employ a partitioned scheme based on the explicit second-order Heun method for the semi-discrete fluid equations and the implicit trapezoidal rule for the structure ODE. While the structure displacement determines the displacement of the equidistant fluid grid nodes, calculation of the mesh velocity obeys the geometric conservation law.

Collecting the fluid unknowns into the vector  $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)^T$ , the DG semi-discretization can be written as a system of ordinary differential equations, if the grid velocity  $\mathbf{w}$  and the grid Jacobian  $J$  are known. This system of ODEs has the form

$$\frac{d}{dt} \mathbf{u} = \mathbf{g}^{\{f\}}(\mathbf{u}, \mathbf{w}, J). \quad (1.157)$$

Furthermore, we may transform the second-order linear ODE (1.153) into a system of first-order ODEs for the vector of unknowns  $\mathbf{q} = (q, \dot{q})^T$ . Thus, we have

$$\frac{d}{dt} \mathbf{q} = \mathbf{g}^{\{s\}}(\mathbf{u}, \mathbf{q}) = \begin{pmatrix} 0 & 1 \\ -\frac{k}{m} & 0 \end{pmatrix} \mathbf{q} + \begin{pmatrix} 0 \\ \frac{A}{m} (p_I(\mathbf{u}) - p_A) \end{pmatrix}. \quad (1.158)$$

These two systems of ODEs are obviously coupled, since the interface pressure  $p_I = p_I(\mathbf{u})$  required in (1.158) is defined by the fluid solution while the arguments  $\mathbf{w}, J$  of the right-hand side of the fluid equations (1.157) are related to the piston position by (1.154).

Time integration is now carried out by the following partitioned Runge-Kutta (RK) scheme with coefficients listed in the Butcher array (1.159), employing explicit discretization of the fluid equations (1.157) and implicit discretization of the structure equations (1.158).

$$\begin{array}{c|cc|cc} c_1 & a_{1,1}^{\{f\}} & a_{1,2}^{\{f\}} & a_{1,1}^{\{s\}} & a_{1,2}^{\{s\}} \\ c_2 & a_{2,1}^{\{f\}} & a_{2,2}^{\{f\}} & a_{2,1}^{\{s\}} & a_{2,2}^{\{s\}} \\ \hline & b_1^{\{f\}} & b_2^{\{f\}} & b_1^{\{s\}} & b_2^{\{s\}} \end{array} = \begin{array}{c|cc|cc} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 & 1/2 & 1/2 \end{array} \quad (1.159)$$

This is a particular implicit-explicit Runge-Kutta (IMEX-RK) scheme which will be discussed in more detail in Section 3.2 regarding the time discretization of advection-diffusion problems. Its precise implementation is given in Algorithm 1.1 which lists one step of this scheme when applied to the coupled system of ODEs resulting from the piston problem. Hereby, the grid Jacobian and mesh velocity needed by the fluid equations are computed from the structure states. While the grid Jacobian is simply obtained by evaluating (1.155) at the current piston position, the computation of the mesh velocity is more elaborate. Using the piston positions both at time  $t^n$  and at an intermediate time corresponding to the fluid stage, Algorithm 1.2 provides the nodal velocities of the DG nodes needed in order to compute the current fluid stage derivative. Regarding this computation, in Algorithm 1.1, Line 4, and Algorithm 1.2, Line 4, we employ the notation

$$a_{3,k}^{\{f\}} = b_k^{\{f\}}, \quad k = 1, 2,$$

for code simplification.

Applying partitioned time integration schemes to coupled problems as in the context of fluid-structure interaction accounts for the fact that the involved subsystems may considerably deviate in terms of their time scales or stiffness properties. Considering the moving piston test case, several forms of mixed time integration approaches have been suggested. In [221], an IMEX-RK approach similar to the above was used, discretizing the fluid and structure equations in time by the same implicit scheme but realizing the coupling of the two physical fields in an explicit way. A similar IMEX procedure was used in [58]. In [161, 160], a multirate scheme referred to as subcycling was used to advance the fluid with small explicit time steps while an implicit scheme was employed to discretize the structure equations using large time steps. More recently, in [25], the moving piston test case was discretized in time by the so-called GARK framework [171] which is a generalized-structure approach to additively partitioned Runge-Kutta methods. The resulting scheme combines IMEX and multirate approaches while coupling of the subproblems is realized both on the level of the discrete time steps and at the level of interior Runge-Kutta stages.

The procedure to compute the mesh velocity as described in Algorithm 1.2 is designed to satisfy the so-called *geometric conservation law (GCL)* in the discrete sense. The GCL is a partial differential equation which has first been formulated by Thomas and Lombard in [193] and basically states that arbitrary mesh motion does not disturb uniform flow. Hence, a corresponding discrete statement is that no disturbances should be introduced by the mesh motion in case of uniform flow, see for instance [129]. The computation of the GCL compliant mesh velocity in Algorithm 1.2 is adapted from the procedure by van Zuijlen and Bijl in [221], where IMEX time integration is applied to the piston problem with finite volume space discretization of the fluid equations. Slight differences between the computation of mesh velocities in Algorithm 1.2 and in [221] are due to the fact that [221] employs implicit time discretization both to the fluid equations and the structure equations, and only handles the coupling procedure explicitly.

In the following numerical experiments, the influence of the fluid discretization on the structure displacement is investigated. Extending the classical moving piston test case, we now assume viscous compressible fluid flow. Therefore, the fluid equations (1.152) are extended to the

**Algorithm 1.1** Partitioned RK step for piston problem

---

**Input:**  $\mathbf{u}^n, \mathbf{q}^n, \Delta t$   
**Output:**  $\mathbf{u}^{n+1}, \mathbf{q}^{n+1}$

- 1: **for**  $j = 1, 2$  **do**
- 2:    $\mathbf{z}^{(j)} \leftarrow \Delta t \mathbf{g}^{\{s\}} \left( \mathbf{u}^n + \sum_{k=1}^2 a_{j,k}^{\{f\}} \mathbf{r}^{(k)}, \mathbf{q}^n + \sum_{k=1}^2 a_{j,k}^{\{s\}} \mathbf{z}^{(k)} \right)$
- 3:    $J^{(j)} \leftarrow \frac{z^{(j)} + L_0}{L_0}$
- 4:    $\mathbf{w}^{(j)} \leftarrow \text{MeshVelocity} \left( \mathbf{q}^n, \mathbf{q}^n + \sum_{k=1}^2 a_{j+1,k}^{\{f\}} \mathbf{r}^{(k)}, \sum_{k=1}^{j-1} a_{j+1,k}^{\{f\}} \mathbf{w}^{(k)}, \Delta t, j \right)$
- 5:    $\mathbf{r}^{(j)} \leftarrow \Delta t \mathbf{g}^{\{f\}} \left( \mathbf{u}^n + \sum_{k=1}^2 a_{j,k}^{\{f\}} \mathbf{r}^{(k)}, \mathbf{w}^{(j)}, J^{(j)} \right)$
- 6: **end for**
- 7:  $\mathbf{u}^{n+1} \leftarrow \mathbf{u}^n + \sum_{j=1}^2 b_j^{\{f\}} \mathbf{r}^{(j)}$
- 8:  $\mathbf{q}^{n+1} \leftarrow \mathbf{q}^n + \sum_{j=1}^2 b_j^{\{s\}} \mathbf{z}^{(j)}$

---

**Algorithm 1.2** MeshVelocity

---

**Input:** structure states  $\mathbf{q}^n, \mathbf{q}^{\{f\}}$ , lower stage velocities  $\mathbf{w}^{\{f\}}, \Delta t$ , stage  $s$   
**Output:** current mesh velocity  $\mathbf{w}$

- 1: Compute old grid nodes by  $\mathbf{x}^n$  by  $x_i^n = -L_0 + (i-1) \frac{q_1^n + L_0}{K}$
- 2: Compute new grid nodes by  $\mathbf{x}^{\{f\}}$  by  $x_i^{\{f\}} = -L_0 + (i-1) \frac{q_1^{\{f\}} + L_0}{K}$
- 3: Transfer  $\mathbf{x}^n, \mathbf{x}^{\{f\}}$  to global vectors of DG nodes  $\mathbf{X}^n, \mathbf{X}^{\{f\}}$  via (1.156)
- 4:  $\mathbf{w} \leftarrow \left( a_{s+1,s}^{\{f\}} \right)^{-1} \left( \frac{1}{\Delta t} (\mathbf{X}^{\{f\}} - \mathbf{X}^n) - \mathbf{w}^{\{f\}} \right)$

---

compressible Navier-Stokes equations

$$\frac{\partial}{\partial t} (J\mathbf{U}) + J \frac{\partial}{\partial x} \hat{\mathbf{F}}(\mathbf{U}, w) = J \frac{\partial}{\partial x} \mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x), \quad (1.160)$$

which is the ALE formulation of the compressible Navier-Stokes equations on fixed grids defined in (1.145). The viscous fluxes  $\mathbf{F}^{visc}(\mathbf{U}, \mathbf{U}_x)$  are again defined by (1.146), where the viscosity coefficient is now set to  $\mu = 0.001$ , whereas the adiabatic coefficient  $\gamma$  and the Prandtl number  $Pr$  are set to  $\gamma = 1.4$  and  $Pr = 0.72$  as in Section 1.4.2.

The further parameters of the moving piston test case are set to

$$m = 0.1, k = 1.0, A = 0.02, p_A = 1.0, L_0 = 1.0,$$

and the initial conditions to

$$\rho_0(x) = 1.0, v_0(x) = 0.0, p_0(x) = 1.0, \text{ for all } x \in \Omega(0), q_0 = 0.0, r_0 = 0.01.$$

With respect to space discretization of the fluid equations, different variants of piecewise linear DG approximation are used – the skew-symmetric variant is combined with the Jameson KEP flux to obtain the KEP-DG scheme, while the standard DG scheme uses a Lax-Friedrichs flux function. Both Legendre-Gauss (LG) and Legendre-Gauss-Lobatto (LGL) nodal distributions are used.

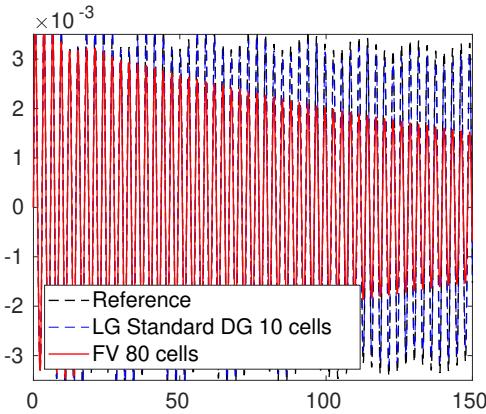


Figure 1.10: Comparison of FV and DG scheme.

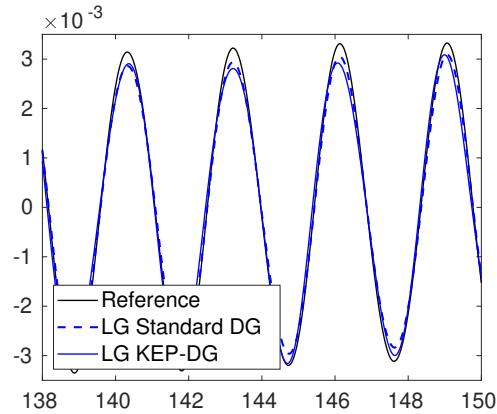


Figure 1.11: KEP-DG scheme vs. standard DG scheme on LG nodes.

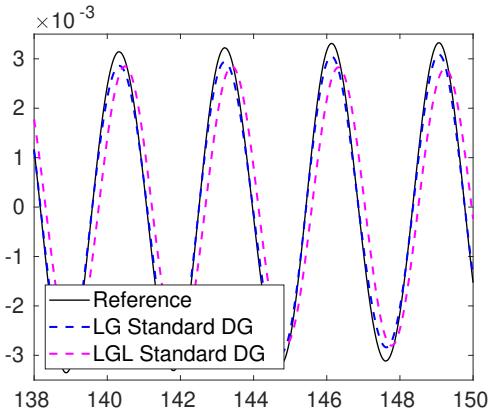


Figure 1.12: Standard DG scheme on LG vs. LGL nodes.

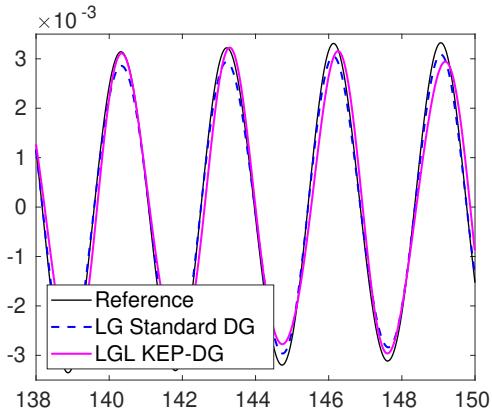


Figure 1.13: KEP-DG scheme on LGL nodes vs. standard DG scheme on LG nodes.

Figures 1.10 to 1.13 show the structural displacement for the various fluid discretizations. The results are furthermore compared to a reference solution which is obtained by the standard fourth-order DG scheme on 500 cells. Furthermore, Figure 1.10 depicts the structural displacement until the final time  $T = 150$  and thereby illustrates the amount of dissipation introduced by a first order finite volume discretization on 80 equidistant cells compared to the second order standard DG scheme on only 10 cells. For a fair comparison, time integration uses the same time step for both spatial discretizations. A higher order scheme is hence mandatory to capture the long term oscillations in the structure. Figure 1.11 shows a comparison of the KEP-DG and the standard DG scheme on LG nodes (again on 10 cells) for a time span from  $t = 138$  to  $t = 150$  with no significant differences. However, for the LGL nodes, Figure 1.12 and Figure 1.13 show considerable differences between the KEP-DG scheme and the standard DG scheme as the frequency of the structure displacement for the kinetic energy preserving variant is closer to the reference solution. The KEP property is thus beneficial regarding the less accurate DG scheme on Legendre-Gauss-Lobatto nodes.

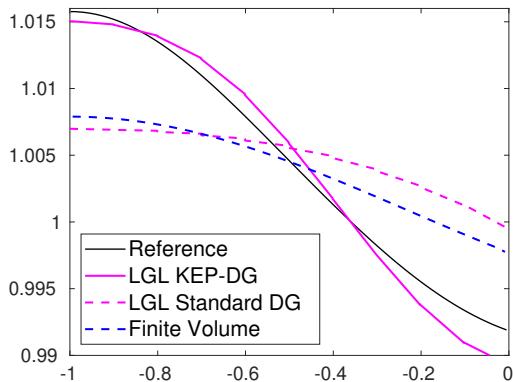


Figure 1.14: LGL DG schemes vs FV scheme.

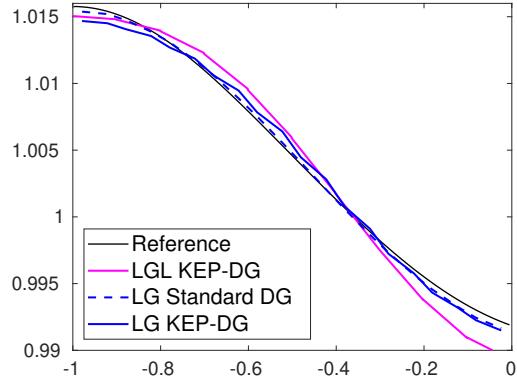


Figure 1.15: LGL KEP-DG scheme vs LG DG schemes.

For the DG scheme on LGL nodes, the increase in accuracy regarding the piston displacement for the KEP variant in comparison to the standard DG scheme has to be attributed to a more accurate fluid approximation in this case. In fact, Figures 1.14 and 1.15 show the pressure distribution with respect to the fixed reference domain  $[-L_0, 0] = [-1, 0]$  at time  $T = 40$  for the investigated DG schemes as well as the finite volume scheme for comparison. In this regard, Figure 1.14 shows that in case of the DG scheme on LGL nodes, the standard DG scheme produces a solution which is closer to the dissipative first order finite volume scheme while the solution of KEP-DG scheme is closer to the reference solution and also to the approximate solution produced by the DG scheme on LG nodes, as shown in Figure 1.15. Since the piston displacement is influenced by the interface pressure, the significant reduction of numerical dissipation introduced to the fluid approximate solution by the KEP-DG scheme transfers to higher accuracy of the approximate structure solution. In addition, Figure 1.15 compares the LGL KEP-DG solution to the approximations produced by the two variants of the DG scheme using the more accurate LG nodes. Here, the results show that both DG variants using LG nodes are more accurate than the LGL KEP-DG scheme. Thus, the higher degree of exactness of the DG quadrature rule pays off for this test case.

## 1.5 Energy conservative approaches for the shallow water equations

The preservation of specific secondary quantities in addition to the primary conserved ones is also of interest in case of the shallow water equations which will be discussed in more detail in Chapter 4 where the focus is put on the simulation of wetting and drying shallow water flows. In fact, while the shallow water equations represent a system of conservation laws consisting of a continuity and a momentum equation which describe the evolution of the water height and the discharge, respectively, we may derive an additional conservation law for the evolution of the total energy which is fulfilled for smooth solutions. However, discretizing the given continuity and momentum equations by conservative schemes does not necessarily

yield a discrete analog of energy conservation. Nonetheless, preventing excessive energy gain or loss may become crucial both from the viewpoint of a theoretical stability analysis and with regard to an accurate simulation of practically relevant fluid flows. For instance, in [79], van't Hof and Veldman argue that respecting both momentum and energy balance is of particular importance for rapidly varied shallow water flows and therefore design a mass, momentum and energy conservative (MaMEC) scheme on staggered grids.

The total energy furthermore represents an entropy function for the shallow water equations and resumes an important role if the exact solution develops discontinuities. In this case, the notion of a solution is extended to weak solutions derived from the integral form of the conservation law and uniqueness is lost. The uniqueness of the physically relevant solution is then recovered by the enforcement of an *entropy inequality*. In particular, for the shallow water equations, the entropy inequality dictates that the total energy should be dissipated across shock discontinuities. A numerical scheme which satisfies a discrete version of the entropy inequality is called *entropy stable* whereas *entropy conservative* schemes satisfy a discrete entropy conservation law and thus a discrete version of an entropy equality. Since Tadmor's pioneering papers [189, 190], entropy conservative schemes have been used as an important building block to devise entropy stable schemes using a comparison principle which roughly states that a finite volume scheme is entropy stable if it contains more numerical diffusion than present in an entropy conservative one, where "more" is to be understood in the sense of ordering between symmetric matrices.

Along this route, in [57], the concept of entropy conservation has been used to construct entropy stable finite volume schemes for the shallow water equations with discontinuous topography. As a second major contribution of [57], a relation has been detected between energy conservation and the preservation of moving water equilibria of the shallow water equations. This is an important aspect since designing numerical schemes which preserve the physically relevant equilibrium states prevents the production of artificial disturbances close to these stationary solutions.

The classical conservative form of the shallow water equations with non-constant bottom topography is given by

$$H_t + (Hv)_x = 0, \quad (1.161)$$

$$(Hv)_t + \left( Hv^2 + \frac{1}{2}gH^2 \right)_x = -gHb_x, \quad (1.162)$$

where  $H$  denotes the water height above the bottom elevation  $b$ , while  $g$  denotes the gravitational constant and  $v$  the flow velocity. The resulting skew-symmetric momentum formulation will be given in Section 1.5.1 and other practically relevant formulations will be discussed in Section 4.1. All stationary solutions of the one-dimensional system of equations (1.161), (1.162) are characterized by the algebraic relations

$$q := Hv \equiv \text{const}, \quad p := \frac{v^2}{2} + g(H + b) \equiv \text{const}, \quad (1.163)$$

where  $q$  and  $p$  are referred to as the *equilibrium variables*. The most important special case of these stationary solution is the so-called *lake at rest* solution with zero flow velocity and

constant water surface, i.e.

$$v \equiv 0, \quad H + b \equiv \text{const.} \quad (1.164)$$

A failure to preserve this particularly relevant stationary solution will most certainly result in severely disturbed simulations of slightly perturbed lake at rest situations, such as tsunami waves in deep ocean, unless extremely fine computational grids are used. Therefore, contemporary schemes for the simulation of shallow water flows are generally designed to be *well-balanced*, i.e. preserve the steady state solution (1.164) due to its significance. The design of numerical schemes preserving the more general moving water equilibria characterized by (1.163) is significantly more difficult. Examples of numerical schemes which preserve these general stationary solutions are the finite volume scheme [142] by Noelle et al. and the DG scheme [211] by Xing which are both based on the technique of hydrostatic reconstruction also briefly discussed in Section 4.3.1. The energy conservative and energy stable numerical schemes constructed by Fjordholm et al. in [57] also preserve the general equilibrium states (1.163) of the one-dimensional shallow water equations but do not require a hydrostatic reconstruction. Thus, the energy conservative approach results in a remarkably simple formulation of an equilibrium preserving scheme. In the following, we will refer to numerical schemes preserving the general steady state solutions (1.163) as being *well-balanced for moving water equilibria* while numerical schemes preserving the special case (1.164) will simply be called well-balanced.

The use of energy conservation to construct well-balanced numerical schemes with or without the inclusion of moving water equilibria will be theoretically analyzed and numerically investigated in the following. First, Section 1.5.1 presents an approach based on the skew-symmetric formulation of the shallow water equations which are discretized in space by the DG scheme on Legendre-Gauss nodes having a generalized SBP property as demonstrated in Section 1.2.1. This scheme is well-balanced without the necessity to use hydrostatic reconstruction in contrast to the schemes discussed in Section 4.3.1 and 4.3.2. However, it is not possible to prove well-balancedness for moving water equilibria for this scheme.

Second, Section 1.5.2 deals with the already mentioned MaMEC scheme [79] by van't Hof and Veldman. This scheme has been constructed with the sole purpose of providing global and local energy conservation in addition to already guaranteed local mass and momentum conservation and the authors do not refer to the issue of well-balancedness in their work. While [79] also compares the MaMEC scheme to a naive finite volume formulation which turns out to disrespect the preservation of the lake at rest equilibrium, the question of provable well-balancedness with or without the inclusion of moving water equilibria remains for this specific scheme. A positive answer regarding simplified staggered grids will be given in Section 1.5.2, proving that the MaMEC scheme is indeed well-balanced for moving water equilibria.

### 1.5.1 A well-balanced and energy conservative DG scheme on Legendre-Gauss nodes for shallow water flow

Analogously to the discrete preservation of the kinetic energy balance in case of the Euler equations as discussed in Section 1.4, we may achieve total energy preservation in the discrete sense for the shallow water equations by using a suitable skew-symmetric form of these

equations and a space-discretization having an SBP property. In this spirit, in [68], the SBP property of the DG scheme on Legendre-Gauss-Lobatto nodes is used to construct a high order entropy preserving numerical method for the shallow water equations in one space dimension which is provably well-balanced regarding still water equilibria, i.e. preserves lake at rest stationary solutions. The considered entropy function is thereby given by the total energy. Furthermore, several variants of entropy stable and well-balanced split-form semi-discretizations based on generalized SBP operators have been designed in [165]. In the following, we present the construction carried out in [153] of an entropy preserving, well-balanced DG scheme on Legendre-Gauss nodes for shallow water flow, based on the generalized SBP property of the DG scheme on Legendre-Gauss nodes as derived in Section 1.2.1. Thus, the construction of entropy preserving well-balanced DG schemes based on a skew-symmetric form of the shallow water equations is not restricted to nodal distributions containing the boundary nodes of DG cells.

### Skew-symmetric formulation of the shallow water equations

Using the product rule, it is possible to derive from the divergence form (1.161), (1.162) of the shallow water equations the skew-symmetric formulation of the momentum equation

$$\frac{1}{2} [(Hv)_t + Hv_t] + \frac{1}{2} [(Hv^2)_x + Hvv_x] + gH(H + b)_x = 0. \quad (1.165)$$

In [68], this formulation is used to derive an entropy preserving, well-balanced DG scheme on Legendre-Gauss-Lobatto nodes using the summation-by-parts property. Hereby, entropy preservation refers to the preservation of total energy  $e$  which represents an entropy function for the shallow water equations. The total energy  $e = k + p$  is composed of the kinetic energy  $k = \frac{1}{2}Hv^2$  and the potential energy  $p = \frac{1}{2}gH^2 + gHb$ .

The DG scheme on Legendre-Gauss nodes uses a quadrature rule of higher degree of exactness but the set of nodes does not include the cell boundaries. However, the same derivations as on [68] can be carried out in order to obtain the desired properties. Given a specific DG cell, we now collect the nodal values of the water height and the discharges into the vectors

$$\begin{aligned} \mathbf{u}_1 &= \mathbf{h} = (H_1, \dots, H_{N+1})^T, \\ \mathbf{u}_2 &= \underline{\mathbf{h}} \mathbf{v} = (H_1 v_1, \dots, H_{N+1} v_{N+1})^T. \end{aligned}$$

Furthermore, the nodal values of the flux functions are given by

$$\begin{aligned} \mathbf{f}_1 &= \underline{\mathbf{h}} \mathbf{v}, \\ \mathbf{f}_2 &= \underline{\mathbf{h}} \mathbf{v}^2 + \frac{g}{2} \mathbf{h}^2. \end{aligned}$$

Discretizing the skew-symmetric formulation (1.161), (1.165) by the DG scheme (1.53) and reusing the respective notation of Section 1.2.1, we obtain

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_1}{dt} + \mathbf{D} \mathbf{f}_1 = \mathbf{M}^{-1} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1, \quad (1.166)$$

$$\frac{\Delta x}{2} \frac{1}{2} \left( \frac{d\mathbf{u}_2}{dt} + \underline{\mathbf{h}} \frac{d\mathbf{v}}{dt} \right) + \frac{1}{2} (\mathbf{D} \underline{\mathbf{h}} \mathbf{v}^2 + \underline{\mathbf{h}} \mathbf{v} \mathbf{D} \mathbf{v}) + g \underline{\mathbf{h}} \mathbf{D}(\mathbf{h} + \mathbf{b}) = \mathbf{M}^{-1} [(k_h - k^*) \mathbf{L}]_{-1}^1, \quad (1.167)$$

where  $k_h$  and  $k^*$  are not yet specified.

Multiplying the above semi-discrete continuity equation by  $\frac{1}{2}\underline{\underline{\mathbf{v}}}$  and adding this to the skew-symmetric momentum equation yields, due to time continuity,

$$\begin{aligned} & \frac{\Delta x}{2} \frac{d\mathbf{u}_2}{dt} + \mathbf{D} \mathbf{f}_2 + \frac{1}{2} \mathbf{s}_{Hv,v} + \frac{g}{2} \mathbf{s}_{H,H} + g \underline{\underline{\mathbf{h}}} \mathbf{D} \mathbf{b} \\ &= \mathbf{M}^{-1} \left( [(k_h - k^*) \mathbf{L}]_{-1}^1 + \frac{1}{2} \underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \right), \end{aligned} \quad (1.168)$$

where  $\mathbf{s}_{Hv,v} = -\mathbf{D} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \underline{\underline{\mathbf{h}}} \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{v} + \underline{\underline{\mathbf{v}}} \mathbf{D} \underline{\underline{\mathbf{h}}} \mathbf{v}$  and  $\mathbf{s}_{H,H} = -\mathbf{D} \mathbf{h}^2 + 2 \underline{\underline{\mathbf{h}}} \mathbf{D} \mathbf{h}$ .

### Mass and momentum balance

Now, the semi-discrete continuity equation (1.166) is obtained by applying the standard DG scheme to the continuity equation (1.161) in divergence form, hence mass preservation is automatically guaranteed. For the momentum balance we consider the additional skew-symmetric discrete terms in equation (1.168) which have the form

$$\mathbf{s}_{\alpha,\beta} = -\mathbf{D} \underline{\underline{\alpha}} \beta + \underline{\underline{\alpha}} \mathbf{D} \beta + \underline{\underline{\beta}} \mathbf{D} \alpha$$

mimicking the product rule. We have

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \mathbf{s}_{\alpha,\beta} &= \mathbf{1}^T \mathbf{M} \left( -\mathbf{D} \underline{\underline{\alpha}} \beta + \underline{\underline{\alpha}} \mathbf{D} \beta + \underline{\underline{\beta}} \mathbf{D} \alpha \right) \\ &= \mathbf{1}^T \left[ (\mathbf{D}^T \mathbf{M} - \mathbf{B}) \underline{\underline{\alpha}} \beta + \underline{\underline{\alpha}} (\mathbf{B} - \mathbf{D}^T \mathbf{M}) \beta + \underline{\underline{\beta}} \mathbf{M} \mathbf{D} \alpha \right] \\ &= -\mathbf{1}^T \mathbf{B} \underline{\underline{\alpha}} \beta + \alpha^T \mathbf{B} \beta. \end{aligned} \quad (1.169)$$

In case of interior node distributions, the boundary matrices  $\mathbf{B}$  are generally not diagonal and  $\mathbf{1}^T \mathbf{M} \mathbf{s}_{\alpha,\beta} \neq \mathbf{0}$ . Therefore, boundary correction terms have to be added to the right-hand side of the DG scheme. Using (1.169), we have

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \mathbf{s}_{Hv,v} &= \mathbf{1}^T \mathbf{M} \mathbf{s}_{v,Hv} = -\mathbf{1}^T \mathbf{B} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \mathbf{v}^T \mathbf{B} \mathbf{f}_1, \\ \mathbf{1}^T \mathbf{M} \mathbf{s}_{H,H} &= -\mathbf{1}^T \mathbf{B} \mathbf{h}^2 + \mathbf{h}^T \mathbf{B} \mathbf{h}. \end{aligned}$$

From (1.168) we derive that for constant bottom topography,  $\mathbf{D} \mathbf{b} = \mathbf{0}$ , the contribution of the volume terms to the change of momentum within a DG cell sums up to

$$\mathbf{1}^T \mathbf{M} \left( \mathbf{D} \mathbf{f}_2 + \frac{1}{2} \mathbf{s}_{Hv,v} + \frac{g}{2} \mathbf{s}_{H,H} \right) = \frac{1}{2} \mathbf{1}^T \mathbf{B} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \frac{1}{2} \mathbf{v}^T \mathbf{B} \mathbf{f}_1 + \frac{g}{2} \mathbf{h}^T \mathbf{B} \mathbf{h}. \quad (1.170)$$

Furthermore, as  $\mathbf{1}^T \left( \frac{1}{2} \underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \right) = \frac{1}{2} \mathbf{v}^T \mathbf{B} \mathbf{f}_1 - \frac{1}{2} [v_h f_1^*]_{-1}^1$ , we may choose

$$k_h = \frac{1}{2} ((Hv^2)_h + g(H_h)^2) = f_{2,h} - \frac{1}{2} (Hv^2)_h + \frac{g}{2} ((H_h)^2 - (H^2)_h), \quad (1.171)$$

$$k^*(\pm 1) = f_2^* - \frac{1}{2} v_h (\pm 1) f_1^*, \quad (1.172)$$

in order to obtain for the surface terms

$$\begin{aligned} & \mathbf{1}^T \left( [(k_h - k^*) \mathbf{L}]_{-1}^1 + \frac{1}{2} \underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \right) \\ &= \frac{1}{2} \mathbf{1}^T \mathbf{B} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \frac{g}{2} \mathbf{h}^T [H_h \mathbf{L}]_{-1}^1 - \left[ f_2^* - \frac{1}{2} v_h f_1^* \right]_{-1}^1 + \frac{1}{2} \mathbf{v}^T \mathbf{B} \mathbf{f}_1 - \frac{1}{2} [v_h f_1^*]_{-1}^1 \\ &= \frac{1}{2} \mathbf{1}^T \mathbf{B} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \frac{g}{2} \mathbf{h}^T \mathbf{B} \mathbf{h} + \frac{1}{2} \mathbf{v}^T \mathbf{B} \mathbf{f}_1 + [f_2^*]_{-1}^1. \end{aligned} \quad (1.173)$$

Comparing (1.170) and (1.173), the change of momentum within a cell directly corresponds to momentum fluxes across the cell boundaries. In addition, the numerical flux  $f_2^*$  is unique on each cell boundary. Therefore, for constant bottom topography, momentum is conserved.

The final form of the skew-symmetric semi-discrete momentum equation is thus given by

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_2}{dt} + \mathbf{D} \mathbf{f}_2 + \frac{1}{2} \mathbf{s}_{Hv,v} + \frac{g}{2} \mathbf{s}_{H,H} + g \underline{\underline{\mathbf{h}}} \mathbf{D} \mathbf{b} = \mathbf{M}^{-1} [(f_{2,h} - f_2^*) \mathbf{L}]_{-1}^1 + \mathbf{M}^{-1} \mathbf{s}^{bc}, \quad (1.174)$$

with the cell boundary correction term

$$\mathbf{s}^{bc} = \frac{1}{2} \underline{\underline{\mathbf{v}}} [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 - \frac{1}{2} \left[ ((Hv^2)_h - v_h f_1^* - g(H_h)^2 + g(H^2)_h) \mathbf{L} \right]_{-1}^1.$$

### Conservation of total energy as an entropy function

As already stated, an entropy function for the shallow water equations is given by the total energy composed of the kinetic energy  $k = \frac{1}{2} Hv^2$  and the potential energy  $p = \frac{1}{2} g H^2 + g H b$ . In the following, we consider these terms separately. The semi-discrete kinetic energy balance can be reconstructed from the initial momentum discretization (1.167) multiplied by  $\underline{\underline{\mathbf{v}}}$ , since  $\frac{d}{dt} \mathbf{k} = \frac{1}{2} (\underline{\underline{\mathbf{v}}} \frac{d\mathbf{u}_2}{dt} + \underline{\underline{\mathbf{u}_2}} \frac{d\mathbf{v}}{dt})$ . We have

$$\frac{\Delta x}{2} \frac{d\mathbf{k}}{dt} + \frac{1}{2} \underline{\underline{\mathbf{v}}} (\mathbf{D} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 + \underline{\underline{\mathbf{h}}} \underline{\underline{\mathbf{v}}} \mathbf{D} \mathbf{v}) + g \underline{\underline{\mathbf{h}}} \underline{\underline{\mathbf{v}}} \mathbf{D}(\mathbf{h} + \mathbf{b}) = \mathbf{M}^{-1} \underline{\underline{\mathbf{v}}} [(k_h - k^*) \mathbf{L}]_{-1}^1.$$

The semi-discrete potential energy balance can be obtained from the semi-discrete continuity equation multiplied by  $g(\underline{\underline{\mathbf{h}}} + \underline{\underline{\mathbf{b}}})$  since  $\frac{d}{dt} \mathbf{p} = g(\underline{\underline{\mathbf{h}}} + \underline{\underline{\mathbf{b}}}) \frac{d\mathbf{h}}{dt}$ . We obtain

$$\frac{\Delta x}{2} \frac{d\mathbf{p}}{dt} + g(\underline{\underline{\mathbf{h}}} + \underline{\underline{\mathbf{b}}}) \mathbf{D} \mathbf{f}_1 = \mathbf{M}^{-1} g(\underline{\underline{\mathbf{h}}} + \underline{\underline{\mathbf{b}}}) [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1.$$

For the total energy, we hence obtain

$$\begin{aligned} & \frac{\Delta x}{2} \frac{d\mathbf{e}}{dt} + \mathbf{D} \left( \frac{1}{2} \underline{\underline{\mathbf{h}}} \mathbf{v}^3 + g \underline{\underline{\mathbf{u}_2}} (\mathbf{h} + \mathbf{b}) \right) + \frac{1}{2} \mathbf{s}_{v,Hv^2} + g \mathbf{s}_{Hv,H+b} \\ &= \mathbf{M}^{-1} \underline{\underline{\mathbf{v}}} [(k_h - k^*) \mathbf{L}]_{-1}^1 + \mathbf{M}^{-1} g(\underline{\underline{\mathbf{h}}} + \underline{\underline{\mathbf{b}}}) [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1. \end{aligned}$$

Considering the cell means, using (1.169), we have

$$\begin{aligned} \mathbf{1}^T \mathbf{M} \frac{\Delta x}{2} \frac{d\mathbf{e}}{dt} &= -\frac{1}{2} \mathbf{v}^T \mathbf{B} \underline{\underline{\mathbf{h}}} \mathbf{v}^2 - g \mathbf{u}_2^T \mathbf{B}(\mathbf{h} + \mathbf{b}) + \mathbf{v}^T [(k_h - k^*) \mathbf{L}]_{-1}^1 \\ &\quad + g(\mathbf{h} + \mathbf{b})^T [(f_{1,h} - f_1^*) \mathbf{L}]_{-1}^1 \\ &= \mathbf{v}^T \left[ \left( \frac{g}{2} (H_h)^2 - k^* \right) \mathbf{L} \right]_{-1}^1 - g(\mathbf{h} + \mathbf{b})^T [f_1^* \mathbf{L}]_{-1}^1. \end{aligned} \quad (1.175)$$

Now, at an interface the sum of ingoing and outgoing fluxes has to be zero. Given such an interface between two cells  $I_{i-1} = [x_{i-1}, x_i]$ ,  $I_i = [x_i, x_{i+1}]$ , we define the jump and the arithmetic mean of a quantity  $a_h$  as

$$[a_h] = a_h^{i+1}(-1) - a_h^i(1), \quad \{a_h\} = \frac{1}{2} (a_h^{i+1}(-1) + a_h^i(1)),$$

respectively. Thus, at an interface, the right-hand side of equation (1.175) provides the terms

$$\begin{aligned} & (\mathbf{v}^{i+1})^T \left( \left( \frac{g}{2} (H_h^{i+1})^2(-1) - k^{*,i+1}(-1) \right) \mathbf{L}(-1) \right) - g(\mathbf{h}^{i+1} + \mathbf{b}^{i+1})^T f_1^* \mathbf{L}(-1) \\ & - (\mathbf{v}^i)^T \left( \left( \frac{g}{2} (H_h^i)^2(1) - k^{*,i}(1) \right) \mathbf{L}(1) \right) + g(\mathbf{h}^i + \mathbf{b}^i)^T f_1^* \mathbf{L}(1) \\ & = \frac{g}{2} [(v_h)(H_h)^2] - f_2^* [v_h] + \frac{1}{2} f_1^* [(v_h)^2] - g f_1^* [(H + b)_h], \end{aligned}$$

where the last equality is obtained by inserting the definition of the auxiliary flux function  $k^*$  given in (1.172).

Furthermore, since  $[a_h b_h] = \{a_h\} [b_h] + [a_h] \{b_h\}$ , and assuming a continuous bottom topography, i.e.  $[b_h] = 0$ , we may rewrite this last expression as

$$\begin{aligned} & g\{v_h\}\{H_h\}[H_h] + \frac{g}{2}\{(H_h)^2\}[v_h] - f_2^*[v_h] + f_1^*\{v_h\}[v_h] - g f_1^*[H_h] \\ & = g(\{v_h\}\{H_h\} - f_1^*)[H_h] + \left( \frac{g}{2}\{(H_h)^2\} - f_2^* + f_1^*\{v_h\} \right) [v_h] \end{aligned}$$

This expression vanishes for the energy conservative numerical flux given by

$$f^* = \begin{pmatrix} f_1^* \\ f_2^* \end{pmatrix} = \begin{pmatrix} \{H_h\}\{v_h\} \\ \{H_h\}\{v_h\}^2 + \frac{g}{2}\{(H_h)^2\} \end{pmatrix}.$$

By using this numerical flux within the nodal DG scheme on Legendre-Gauss nodes, an entropy conserving scheme for the shallow water equations is constructed.

## Well-balancedness

Well-balancedness for lake at rest situations  $v_h \equiv 0$  and  $H_h + b_h \equiv const$  and a continuous discrete bottom topography can also be proven for the entropy conserving DG scheme on Legendre-Gauss nodes. We need to show that these lake at rest steady state solutions are discretely preserved. Since the velocity for the still water steady state is  $v_h \equiv 0$ , the continuity equation (1.166) directly reduces to stationary water height, i.e.  $\frac{\Delta x}{2} \frac{d\mathbf{u}_1}{dt} = 0$ . The momentum equation (1.174) yields

$$\begin{aligned} \frac{\Delta x}{2} \frac{d\mathbf{u}_2}{dt} &= -g \underline{\mathbf{h}} \mathbf{D}(\mathbf{h} + \mathbf{b}) + \mathbf{M}^{-1} \left[ \left( \frac{g}{2} (H_h)^2 - f_2^* \right) \mathbf{L} \right]_{-1}^1 \\ &= \mathbf{M}^{-1} \left[ \left( \frac{g}{2} (H_h)^2 - \frac{g}{2} \{(H_h)^2\} \right) \mathbf{L} \right]_{-1}^1. \end{aligned}$$

Now, since we assume a continuous discrete bottom topography, hence  $[b_h] = 0$ , the lake at rest condition  $[H_h + b_h] = 0$  yields  $[H_h] = 0$  and thus

$$\left[ \left( \frac{g}{2} (H_h)^2 - \frac{g}{2} \{(H_h)^2\} \right) \mathbf{L} \right]_{-1}^1 = \left[ \left( \frac{g}{2} (H_h)^2 - \frac{g}{2} (H_h)^2 \right) \mathbf{L} \right]_{-1}^1 = 0.$$

Hence we obtain  $\frac{\Delta x}{2} \frac{d\mathbf{u}_2}{dt} = 0$ , proving that the lake at rest situation is preserved.

### Violation of well-balancedness for moving water equilibria

For the skew-symmetric DG scheme on Legendre-Gauss nodes constructed in this section, the preservation of moving water equilibria already fails for the momentum equation.

In fact, if  $Hv \equiv const$ , then  $\mathbf{f}_1$  contains constant nodal values and thus  $\mathbf{D}\mathbf{f}_1 = \mathbf{0}$ . The semi-discrete continuity equation (1.166) thus yields

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_1}{dt} = \mathbf{M}^{-1}[(Hv)_h - \{H_h\}\{v_h\}] \mathbf{L}_{-1}^1 \quad (1.176)$$

However, even though  $q = (Hv)_h$  is constant and thus continuous over cell boundaries,  $H_h$  and  $v_h$  or at least one of these representations in DG space generally admit jumps at these points. Therefore, we generally have  $q \neq \{H_h\}\{v_h\}$  implying that the right-hand side of (1.176) does not vanish unless the polynomial degree of the DG space is  $N = 0$ .

One might argue that the lack of well-balancedness for moving water equilibria probably results from the exclusion of the cell boundary nodes from the DG nodal set. Thus, we next consider the skew-symmetric DG formulation on Legendre-Gauss-Lobatto (LGL) nodes, assume a continuous bottom topography and construct an initial solution by evaluating an equilibrium solution at the DG nodes such that

$$\underline{\underline{\mathbf{h}}}\mathbf{v} = const, \quad \frac{1}{2}\mathbf{v}^2 + g(\mathbf{h} + \mathbf{b}) = const$$

for the initial state. Thereby, we obtain unique values at the cell boundaries. Due to the resulting initial continuity of  $H_h$  and  $v_h$ , the right-hand side of (1.176) vanishes at least for  $t = 0$ . Furthermore the cell boundary contributions in the skew-symmetric momentum equation (1.174) cancel out with  $\mathbf{s}^{bc} = \mathbf{0}$  and  $f_{2,h} - f_2^* = 0$ . Considering the remaining volume terms, since we have  $\mathbf{s}_{Hv,v} = \mathbf{0}$  due to  $(Hv)_h$  being constant, the momentum equation for  $t = 0$  results in

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_2}{dt} = -\mathbf{D}\underline{\underline{\mathbf{h}}}\mathbf{v}^2 - g\underline{\underline{\mathbf{h}}}\mathbf{D}(\mathbf{h} + \mathbf{b}) = -\underline{\underline{\mathbf{h}}}(\underline{\mathbf{v}}\mathbf{D}\mathbf{v} - g\mathbf{D}(\mathbf{h} + \mathbf{b})) . \quad (1.177)$$

Since in general, we have  $\underline{\mathbf{v}}\mathbf{D}\mathbf{v} \neq \frac{1}{2}\mathbf{D}\mathbf{v}^2$ , the right-hand side of equation (1.177) does not vanish.

Therefore, we may not expect the skew-symmetric DG schemes to respect moving water equilibria. In fact, for the DG scheme on LGL nodes we can simply measure the right-hand side of (1.177) for the widely used classical test case regarding a moving water stationary state described for instance in [142, 57, 211]. For this test case on the spatial domain  $\Omega = [0, 25]$ , the gravitational constant is set to  $g = 9.812$  and the bottom topography is described by the function

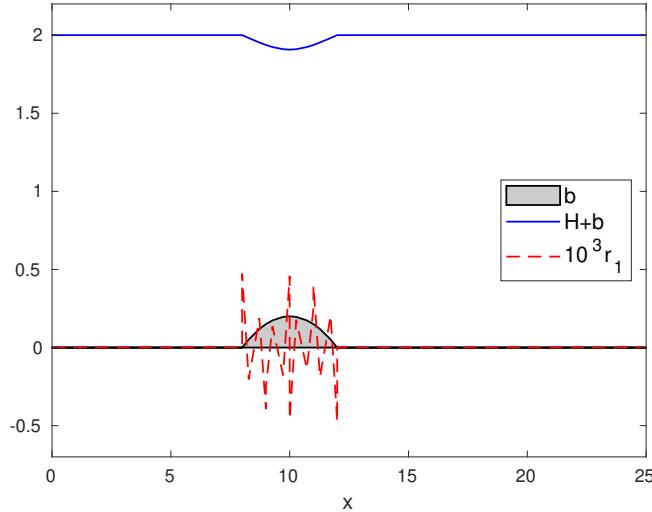
$$b(x) = \begin{cases} 0.2 - 0.05(x - 10)^2 & \text{if } |x - 10| \leq 2, \\ 0 & \text{otherwise.} \end{cases} \quad (1.178)$$

Furthermore, initial solutions for the water height  $H$  and the discharge  $Hv$  are computed at the LGL nodes using the following values for the equilibrium variables,

$$q = Hv = 4.42, \quad p = \frac{v^2}{2} + g(h + b) = 22.06605, \quad (1.179)$$

	$N = 3$	$N = 4$	$N = 5$	$N = 6$	$N = 7$
$\max_{i=1}^K \ \mathbf{r}_1\ _\infty$	4.73e-04	1.10e-04	4.84e-06	1.98e-06	1.55e-07
$\max_{i=1}^K \ \mathbf{r}_2\ _\infty$	1.78e-14	3.55e-14	8.53e-14	1.95e-13	2.25e-13

Table 1.5: Magnitude of the terms (1.180) on LGL nodes.

Figure 1.16: Initial equilibrium solution and nodal values of  $\mathbf{r}_1$  defined in (1.180) for the DG scheme on LGL nodes and  $N = 3$ .

where Newton iteration is applied in order to solve for the initial nodal values of  $H$  as described for instance in [142]. For the DG space discretization, the computational domain  $\Omega$  is hereby divided into 125 cells. For comparison, the terms

$$\mathbf{r}_1 = \underline{\mathbf{v}} \mathbf{D} \mathbf{v} + g \mathbf{D}(\mathbf{h} + \mathbf{b}) \quad \text{and} \quad \mathbf{r}_2 = \mathbf{D} \left( \frac{1}{2} \mathbf{v}^2 + g(\mathbf{h} + \mathbf{b}) \right) \quad (1.180)$$

are now evaluated on each DG cell, where  $\mathbf{D}$  is the first-derivative operator on LGL nodes. In Table 1.5, the maximum norm of the vectors  $\mathbf{r}_1, \mathbf{r}_2$ , taken over the nodes of all DG cells, is listed for polynomial degrees  $N = 3, \dots, 7$ . As we can see, the magnitude of the respective term on the right hand side of (1.177) is generally small and decreasing with increasing order of the DG scheme but is not on the level of machine accuracy. From Figure 1.16, depicting the initial solution and the nodal values of  $\mathbf{r}_1$  for the  $DG(N = 3)$  scheme on LDG nodes, we furthermore deduce that the largest absolute values of  $\mathbf{r}_1$  are assumed in the regions of non-flat bottom topography. In contrast, the components of the vector  $\mathbf{r}_2$  are in the range of machine accuracy indicating that the constructed initial solution is in fact an equilibrium solution.

### 1.5.2 Well-balancedness of the energy conservative MaMEC scheme

The MaMEC scheme [79] is a structure-preserving finite volume type scheme which is mass, momentum and energy conservative. Furthermore, as highlighted in [57], there is a close relationship between energy conservation and the preservation of general equilibrium states of the shallow water equations. However, to the author's knowledge, an investigation of the MaMEC scheme in one space dimension regarding its potential well-balancedness properties for moving water equilibria has not yet been undertaken. In addition, the characterization of energy conservative finite volume schemes in [57] does not directly translate to the MaMEC scheme since this method operates on staggered grids using alternate grid points for the water height on the one hand and the velocity and discharge on the other hand. Hereby, the scheme is allows for non-uniform grids, i.e. it is not necessary that one set of nodes lies halfway the other set of nodes.

In order to construct the MaMEC scheme in one space dimension, two grids need to be defined. For this purpose, we consider given grid points  $x_i$ ,  $i = 1, \dots, K$ , not necessarily equidistant, with  $x_i < x_{i+1}$ . We then construct finite volume cells around these nodes with left and right cell boundary points denoted by  $x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}$ , respectively. This second set of nodes constitutes a dual grid with nodes  $x_{i+\frac{1}{2}}$ ,  $i = 0, \dots, K$ . Specific distances needed for the definition of the MaMEC scheme are given by the lengths of the finite volume and the dual cells, i.e.

$$\Delta x_{i+\frac{1}{2}} = x_{i+1} - x_i, \quad \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}. \quad (1.181)$$

In addition, we need the distances between adjacent alternate grid points, given by

$$\delta x_{i+\frac{1}{4}} = x_{i+\frac{1}{2}} - x_i, \quad \delta x_{i+\frac{3}{4}} = x_{i+1} - x_{i+\frac{1}{2}}. \quad (1.182)$$

The unknowns of the MaMEC semi-discretization of the shallow water equations are specified by the nodal values of the water height  $H$  at the nodes  $x_i$  and the velocity  $v$  at the dual nodes  $x_{i+\frac{1}{2}}$ . Furthermore, the bottom topography  $b$  is computed at the primary nodes  $x_i$  analogous to the water height.

The semi-discrete continuity equation is now given by

$$\frac{dH_i}{dt} + \frac{q_{i+\frac{1}{2}} - q_{i-\frac{1}{2}}}{\Delta x_i} = 0, \quad (1.183)$$

where the discharges  $q_{i\pm\frac{1}{2}}$  at the dual grid points are defined by

$$q_{i\pm\frac{1}{2}} = v_{i\pm\frac{1}{2}} \{H\}_{i\pm\frac{1}{2}}, \quad \text{with} \quad \{H\}_{i\pm\frac{1}{2}} := \frac{H_i + H_{i\pm 1}}{2}. \quad (1.184)$$

As shown in [79], the semi-discrete continuity equation (1.183) on the primary grid now translates to an analogous finite volume semi-discretization on the dual grid of the form

$$\frac{dH_{i+\frac{1}{2}}}{dt} + \frac{q_{i+1} - q_i}{\Delta x_{i+\frac{1}{2}}} = 0, \quad (1.185)$$

with

$$H_{i+\frac{1}{2}} = \frac{\delta_{i+\frac{1}{4}} H_i + \delta_{i+\frac{3}{4}} H_{i+1}}{\Delta x_{i+\frac{1}{2}}}, \quad q_i = \frac{\delta_{i-\frac{1}{4}} q_{i+\frac{1}{2}} + \delta_{i+\frac{1}{4}} q_{i-\frac{1}{2}}}{\Delta x_i}. \quad (1.186)$$

The momentum equation is furthermore discretized on the dual grid by

$$\frac{d \left( H_{i+\frac{1}{2}} v_{i+\frac{1}{2}} \right)}{dt} + \frac{FV_{i+1} - FV_i}{\Delta x_{i+\frac{1}{2}}} = -g \{H\}_{i+\frac{1}{2}} \frac{b_{i+1} - b_i}{\Delta x_{i+\frac{1}{2}}}, \quad (1.187)$$

where the finite volume type discrete flux  $FV_i$  is defined by

$$FV_i = q_i \frac{v_{i+\frac{1}{2}} + v_{i-\frac{1}{2}}}{2} + g \frac{H_i^2}{2}. \quad (1.188)$$

In the context of investigating well-balancedness, boundary conditions are usually neglected. We therefore only consider periodic problems and realize periodic boundary conditions by setting  $x_{K+1} = x_1$ ,  $x_{K+\frac{1}{2}} = x_{\frac{1}{2}}$ . Regarding the above MaMEC scheme for the shallow water equations in one space dimension, we now have the following result on well-balancedness for moving water equilibria.

**Lemma 1.19.** *The MaMEC scheme (1.183), (1.187) is well-balanced for moving water equilibria. More precisely, if*

$$q_{i+\frac{1}{2}} \equiv C_1 \text{ and } p_i = \frac{v_{i+\frac{1}{2}} v_{i-\frac{1}{2}}}{2} + g(H_i + b_i) \equiv C_2$$

for all  $i$ , with constants  $C_1$  and  $C_2$ , then

$$\frac{dH_i(t)}{dt} = \frac{d(Hv)_{i+\frac{1}{2}}(t)}{dt} = 0, \quad \text{for all } t \geq 0,$$

on all grid nodes. Hence, the scheme exactly preserves general equilibrium states of the shallow water equations in one space dimension including moving water stationary solutions.

*Proof.* Concerning the continuity equation, by inserting the assumption  $q_{i+\frac{1}{2}} = q_{i-\frac{1}{2}} = C_1$  into the equation (1.183) we directly obtain  $\frac{dH_i}{dt} \equiv 0$ . Constancy of  $q_{i+\frac{1}{2}}$  also yields constancy of the discharges at the primary nodes, as by the definition of  $q_i$  in (1.186) and the definition of the distances in (1.181) and (1.182), we have

$$q_i = C_1 \frac{\delta_{i-\frac{1}{4}} + \delta_{i+\frac{1}{4}}}{\Delta x_i} = C_1, \quad \text{for } i = 1, \dots, K.$$

Therefore, the momentum equation (1.187) expands to

$$\frac{d(Hv)_{i+\frac{1}{2}}}{dt} + C_1 \frac{v_{i+\frac{3}{2}} - v_{i-\frac{1}{2}}}{2\Delta x_{i+\frac{1}{2}}} + g \frac{H_{i+1}^2 - H_i^2}{2\Delta x_{i+\frac{1}{2}}} = -g \{H\}_{i+\frac{1}{2}} \frac{b_{i+1} - b_i}{\Delta x_{i+\frac{1}{2}}}.$$

Using  $\frac{1}{2}(H_{i+1}^2 - H_i^2) = \{H\}_{i+\frac{1}{2}}(H_{i+1} - H_i)$ , this further simplifies to

$$\frac{d(Hv)_{i+\frac{1}{2}}}{dt} = -C_1 \frac{v_{i+\frac{3}{2}} - v_{i-\frac{1}{2}}}{2\Delta x_{i+\frac{1}{2}}} - g\{H\}_{i+\frac{1}{2}} \left( \frac{H_{i+1} + b_{i+1} - H_i - b_i}{\Delta x_{i+\frac{1}{2}}} \right).$$

Substituting  $C_1 = q_{i+\frac{1}{2}} = (\{H\}v)_{i+\frac{1}{2}}$  and rearranging, we obtain

$$\begin{aligned} \frac{d(Hv)_{i+\frac{1}{2}}}{dt} &= -\frac{\{H\}_{i+\frac{1}{2}}}{\Delta x_{i+\frac{1}{2}}} \left( \frac{v_{i+\frac{1}{2}}v_{i+\frac{3}{2}}}{2} + g(H_{i+1} + b_{i+1}) - \left( \frac{v_{i-\frac{1}{2}}v_{i+\frac{1}{2}}}{2} + g(H_i + b_i) \right) \right) \\ &= -\frac{\{H\}_{i+\frac{1}{2}}}{\Delta x_{i+\frac{1}{2}}} (p_{i+1} - p_i) = 0, \end{aligned}$$

where in the last equality, we used the assumption  $p_{i+1} = p_i = C_2$  on the constancy of the discrete equilibrium variable  $p_i$ . Hence, the MaMEC scheme in one space dimension preserves general equilibrium states.  $\square$

**Remark 1.20.** *The values  $q_{i+\frac{1}{2}}$  and  $p_i$  which are assumed to be constant in Lemma 1.19 represent the discrete equilibrium variables with continuous counterparts defined in (1.163). Hereby, both  $q_{i+\frac{1}{2}}$  and  $p_i$  depend on water height and velocity values on the respective other grid and therefore assume a staggered form depending on a slightly larger stencil. We note, that a similar trait is found in the well-balancedness property of the energy conservative finite volume schemes by Fjordholm et al. [57]. In that work, the discrete equilibrium variable  $q$  considered for well-balancedness for moving water equilibria is taken as the staggered momentum  $q_{i+\frac{1}{2}} = \{H\}_{i+\frac{1}{2}}\{v\}_{i+\frac{1}{2}}$ . We may therefore assume that some kind of staggered definition of the discrete equilibrium variables is necessary in order to preserve moving water equilibria without special design features like hydrostatic reconstruction.*

For the same test case as studied in Section 1.5.1, we now compute solutions with the MaMEC scheme on a uniform staggered grid, using the uniform cell length  $\Delta x$ .

Hereby, the computational domain  $\Omega = [0, 25]$  is discretized by  $K$  primary grid points for the water height, denoted by  $x_i = (i - \frac{1}{2})\Delta x$ ,  $i = 1, \dots, K$ , and  $K + 1$  dual grid points for the velocity, given by  $x_{i+\frac{1}{2}} = i\Delta x$ ,  $i = 0, \dots, K$ , where  $\Delta x = \frac{|\Omega|}{K}$ . The bottom topography and initial conditions in terms of the equilibrium variables are again given by (1.178) and (1.179), respectively, and periodic boundary conditions are implemented. The simulation is run until final time  $T = 5$  using the third order Shu-Osher TVD Runge-Kutta scheme developed in [177] for time integration with time step  $\Delta t = \frac{1}{200}$ . Table 1.6 presents the errors in maximum norm of the nodal values of  $H$  and  $v$ , respectively, for successively refined grids. The table clearly shows that the MaMEC scheme is well-balanced. In addition, Figure 1.17 depicts the numerical solution at time  $T = 5$  using the MaMEC scheme on  $K = 125$  primary grid points. The numerical solution is clearly indistinguishable from the initial equilibrium state as predicted by the theoretical investigation of well-balancedness for moving water equilibria.

	$N = 10$	$K = 50$	$K = 100$	$K = 200$	$K = 400$
$err_H$	2.22e-16	2.22e-16	2.22e-16	2.66e-15	4.00e-15
$err_v$	4.44e-16	4.44e-16	8.88e-16	7.99e-15	7.99e-15

Table 1.6: Well-balancedness of MaMEC scheme: errors in maximum norm for water height and velocity.

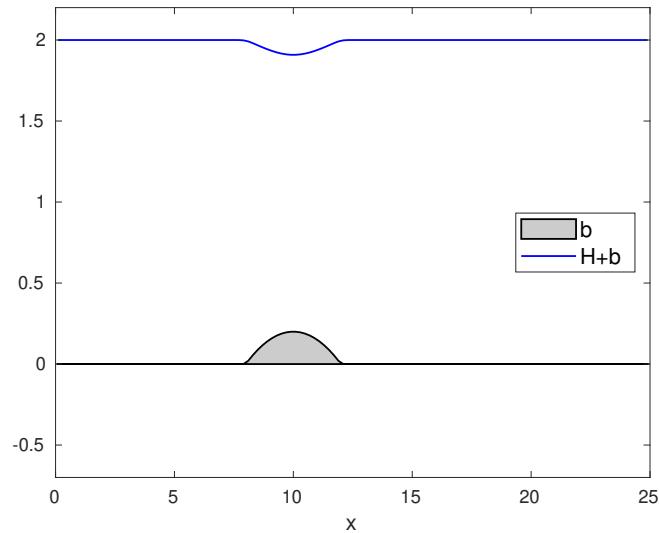


Figure 1.17: Numerical solution of the moving water well-balancedness test case at time  $T = 5$  produced by the MaMEC scheme with uniform cell length  $\Delta x = 0.2$ .

## Chapter 2

# Viscous Flow

### *Discretizing Diffusion Terms in DG Framework*

Various approaches to discretize diffusion terms within a DG scheme have been introduced in the literature since the discretization of higher order spatial derivatives is less natural than for first order derivatives. In fact, the original introduction of the DG approach was aimed at the numerical approximation of first order hyperbolic conservation laws as an alternative to the widely used finite volume approaches.

Therefore, at first sight, although quite natural for advection operators, the inherent discontinuity of the DG approximate solution does not offer any intrinsic way to discretize diffusion operators. When computing the diffusion flux at cell interfaces, neither a unique solution value nor unique derivatives are available. However, several techniques to compute diffusion fluxes within a discontinuous Galerkin framework have been successfully employed, either using specially designed penalty terms within a finite element approach as in [5, 17] or by rewriting equations of convection-diffusion type into a system of first order equations using auxiliary variables for the solution derivatives as in [13, 44, 14, 6]. Among these schemes, the first method by Bassi and Rebay [13] is the first extension of the DG scheme to the compressible Navier-Stokes equations and is usually termed the *BR1 scheme*. It is based on rewriting the equations containing viscous terms into a larger, extended first-order degenerate system of PDEs with the gradient as a new unknown. After this reformulation, the standard DG approach is applied to the extended system which necessitates to prescribe two types of numerical fluxes. Hereby, the BR1 scheme is the simplest approach, using arithmetic means for both types of fluxes.

Motivated by the successful numerical results obtained with the BR1 scheme, Cockburn and Shu [44] analyzed various methods based on the reformulation into a first-order PDE and identified the class of *local discontinuous Galerkin (LDG) methods*. For this, they derived conditions on the numerical fluxes to guarantee stability, convergence and a suboptimal error estimate of order  $N$  when using an approximation space of polynomial degree  $N$ . The analysis by Cockburn and Shu shows suboptimal convergence of the BR1 scheme for odd  $N$  while the choice of alternating numerical fluxes usually associated with the LDG scheme lead to optimal convergence of order  $N + 1$ . Some further disadvantageous properties for purely

elliptic problems such as a widened stencil and lack of stability have been attributed to the BR1 scheme in [6]. However, being simple to code, parameter-free, and generic for non-linear viscous fluxes and arbitrary grids, the BR1 scheme is still considered an attractive approach and has recently been re-investigated in Gassner et al. [67], where the neutral behavior of BR1 with respect to artificial dissipation over element interfaces and the resulting stability for the compressible Navier-Stokes equations has been proven. In [15], Bassi and Rebay introduced a second approach to the discretization of viscous terms within the DG framework, termed the *BR2 scheme*, which modifies the BR1 approach to yield a more compact stencil suitable for efficient implicit time integration. The BR2 diffusion discretization includes a penalty parameter which determines stability and accuracy of the scheme. The above methods using a reformulation into a first order system were then analyzed by Arnold et al. [6] in a unified framework also including the early penalty approaches [5, 17]. For this unified analysis, Arnold et al. rewrote the schemes into a primal formulation by eliminating the introduced auxiliary variable. Related to the early penalty methods is the so-called  $(\sigma, \mu)$ -family of DG diffusion discretizations which has been revisited more recently by van Leer et al. in [109] and will be discussed in Section 2.2.

Initially, the focus of the above approaches concentrated on developing stable and accurate DG discretizations of diffusion terms. Subsequently, particular attention was paid to increasing the efficiency of these schemes also considering the aspect of time integration.

On the one hand, by successively reducing the amount of coupling between the degrees of freedom on adjacent DG cells, efficiency was increased – particularly in view of implicit time integration and implementation in parallel hardware environment. For this purpose, based on the LDG scheme, Peraire and Persson developed the *compact DG (CDG) scheme* [157, 158] increasing compactness of the formulation in the multi-dimensional case. With particular regard to efficient implicit time integration, the class of *hybridizable discontinuous Galerkin (HDG) schemes* which has been reviewed in [140] locally projects the numerical solution to the element boundaries resulting in a final system with significantly less globally coupled unknowns. On the other hand, efficiency may be increased by compensating the lack of continuity of the DG approximation since for diffusion problems, standard DG schemes are known to be more time-consuming than classical finite element schemes, see [120]. Examples are the *recovery-based DG method* by van Leer et al. [110] which recovers a smooth approximation and increases the accuracy of the DG scheme, as well as the *reconstructed DG method* by Luo et al. [120] which locally reconstructs a polynomial approximation to the DG solution across element boundaries and computes the diffusion fluxes based on this smooth approximation.

The recovery-based and the reconstructed DG scheme are formulated without rewriting viscous equations into first order systems. Other DG diffusion discretizations using this approach are the method by Gassner et al. [66] constructing diffusion fluxes based on the solution of diffusive generalized Riemann problems and the direct discontinuous Galerkin (DDG) method introduced by Liu and Yan [117, 116] which can be viewed as a multi-term penalty method.

### Organization of this chapter

This chapter concentrates on specific aspects regarding the discretization of diffusion terms in the DG framework. First, in Section 2.1, for the one-dimensional linear diffusion equation, we

review those contemporary approaches which are based on a reformulation into a first-order system, namely the LDG and BR schemes. In this context, the upwind SBP properties of some of these schemes will be investigated more closely. Thereafter, Section 2.2 deals with  $(\sigma, \mu)$ -family of DG diffusion discretizations. For DG schemes in one space dimension on Legendre-Gauss-Lobatto nodes, we show that both the BR1 scheme and the BR2 scheme for any value of the penalty parameter may be understood as  $(\sigma, \mu)$ -schemes. Furthermore, we prove certain properties of the  $(\sigma, \mu)$ -family regarding symmetry and dissipativity under conditions on the parameters  $\sigma$  and  $\mu$ . Finally, considering the BR2 flux for the DG scheme on Legendre-Gauss-Lobatto nodes, the lifting operator may be calculated either by exact projection or using inexact numerical integration on the given nodes. Both versions are brought together in Section 2.3 by extending recent results by Quaeghebeur et al. [162] on the equivalence of the BR2 scheme and the classical interior penalty formulation for linear diffusion in one space dimension.

Subsequently, in Section 2.4, we study the influence on dissipation and dispersion properties of the two most frequently used alternating versions of the LDG scheme as well as the BR schemes. The analysis highlights a significant difference between the two possible ways to choose the alternating LDG fluxes. Furthermore, we will detect an odd-even phenomenon regarding the accuracy of the different DG diffusion discretizations for well-resolved problems and differences of the wave propagation properties for DG schemes on either Legendre-Gauss nodes or on Legendre-Gauss-Lobatto nodes.

## 2.1 DG discretization of linear diffusion in one space dimension

To simplify the presentation, we consider the linear heat equation

$$\frac{\partial}{\partial t} U(x, t) = d \frac{\partial^2}{\partial x^2} U(x, t), \quad (x, t) \in Q = \Omega \times (0, T), \quad \Omega = (x_\alpha, x_\beta) \quad (2.1)$$

with diffusion coefficient  $d > 0$ , supplemented by the periodic initial condition  $U(x, 0) = U_0(x)$  in  $L^2(\Omega)$  and periodic boundary conditions. In the above definition of the linear heat equation, we now use the upper case letter  $U$ , different from the use of lower case  $u$  within the scalar partial differential equations introduced in Chapter 1. This change in notation allows to simplify the variational DG formulation of (2.1) by using  $u$  to indicate the DG approximation instead of  $u_h$  as in Section 1.2.1.

The DG discretization of (2.1) is now obtained as follows. First, the computational domain  $\Omega$  is again partitioned into cells denoted by  $I_j = (x_j, x_{j+1})$ ,  $j = 1, \dots, E$  with  $x_1 = x_\alpha$ ,  $x_{E+1} = x_\beta$ . In the following, we shall consider uniform grids with cell length  $x_{j+1} - x_j = \Delta x$ . Analogously to the construction of DG schemes in one space dimension in Section 1.2.1, basis functions and test functions are taken from the finite element space

$$V_h = \left\{ v \in L^2(\Omega) \mid v|_{I_j} \in \mathcal{P}_N(I_j), \quad \forall j = 1, \dots, E \right\}, \quad (2.2)$$

where  $\mathcal{P}_N(I_j)$  denotes the space of polynomial functions on  $I_j$  of degree at most  $N$ . Herein, in order to simplify the notation in the current section, the approximate solution of the

DG scheme and the test functions do not carry the subscript  $h$ . As already discussed in Section 1.2.1, the functions in  $V_h$  may be discontinuous across element boundaries. In this chapter, at an element boundary given by  $x_j$ , the left-hand side and right-hand side values of a piece-wise continuous function  $v$  are denoted by  $v_j^- = v^-(x_j)$  and  $v_j^+ = v^+(x_j)$ , respectively. The corresponding jump at element interfaces is denoted by  $[v]_j = v_j^+ - v_j^-$  and the arithmetic mean is given by  $\{v\}_j = \frac{1}{2}(v_j^+ + v_j^-)$ . Furthermore, periodicity is realized by setting  $v_1^- = v_{E+1}^-$  and  $v_{E+1}^+ = v_1^+$ .

The BR1, BR2 and the LDG scheme are all derived from the following first order reformulation of the linear heat equation (2.1) given by

$$\frac{\partial}{\partial t} U(x, t) = d \frac{\partial}{\partial x} Q(x, t), \quad Q(x, t) = \frac{\partial}{\partial x} U(x, t), \quad (2.3)$$

with an auxiliary variable  $Q$ . The corresponding element-wise DG space discretization to obtain both the approximate solution  $u(t) \in V_h$  and the auxiliary variable  $q(t) \in V_h$  is then given by

$$(u_t, v)_j = d \left( -(q, v_x)_j + q_{j+1}^* v_{j+1}^- - q_j^* v_j^+ \right), \quad \forall v \in V_h, \quad \forall j = 1, \dots, E, \quad (2.4)$$

$$(q, r)_j = -(u, r_x)_j + u_{j+1}^* r_{j+1}^- - u_j^* r_j^+, \quad \forall r \in V_h, \quad \forall j = 1, \dots, E, \quad (2.5)$$

where  $(\cdot, \cdot)_j$  denotes the usual inner product in  $L^2(I_j)$  and where  $q^*$  and  $u^*$  represent suitable numerical fluxes determining the chosen DG diffusion scheme.

Regarding the specification of the numerical fluxes, the simplest approach is the BR1 scheme given by the choice of arithmetic means, i.e.

$$q_j^{*,BR1} = \{q\}_j, \quad u_j^{*,BR1} = \{u\}_j. \quad (2.6)$$

Furthermore, the original LDG scheme [44] yields a parameter-dependent family of diffusion fluxes

$$q_j^{*,LDG} = \{q\}_j - c_{12}[q]_j + c_{11}[u]_j, \quad u_j^{*,LDG} = \{u\}_j + c_{12}[u]_j, \quad (2.7)$$

which, in one space dimension, contains the BR1 approach as a specific case with  $c_{11} = c_{12} = 0$ . Here, we only consider the common choice of alternating LDG fluxes with  $c_{12} = \pm 1$ ,  $c_{11} = 0$ , which offers two different variants. One implementation is thus given by

$$q_j^{*,LDG_a} = q_j^-, \quad u_j^{*,LDG_a} = u_j^+, \quad (2.8)$$

which uses opposite wind direction compared to the upwind advective flux in (2.4) and is therefore termed as inconsistent with the advective flux by Cheng and Shu [40]. The second variant is specified by

$$q_j^{*,LDG_b} = q_j^+, \quad u_j^{*,LDG_b} = u_j^-, \quad (2.9)$$

and termed consistent with the advective flux.

The BR2 scheme is a modification of the BR1 approach which was first suggested in [15] in order to obtain a more compact stencil suitable for efficient implicit time integration since the

BR2 stencil contains only immediate neighbors. In the general case, this is in contrast to the wider stencils of BR1 scheme. However, in one space dimension and integrated into a nodal DG scheme on Legendre-Gauss-Lobatto nodes, the stencils of BR1 and BR2 are the same.

For the BR2 approach, the numerical flux in the auxiliary equation (2.5) equals the choice for BR1, i.e.  $u^{*,BR2} = \{u\}$ . Furthermore,  $q^{*,BR2}$  is determined by local lifting operators  $l_j$  which lift the jumps  $[u]_j$  into the DG approximation space. More precisely, we calculate  $l_j([u]) \in V_h$  based on the projection property

$$(l_j([u]), v) = [u]_j \{v\}_j, \quad \forall v \in V_h, \quad (2.10)$$

where  $(\cdot, \cdot)$  denotes the classical  $L^2$  inner product on  $V_h$ , i.e.  $(v, w) = \sum_{j=1}^E (v, w)_j$ .

The numerical flux  $q^*$  in (2.4) is then defined by

$$q_j^* = q_j^{*,BR2} = \{u_x + \eta_e l_j([u])\}_j, \quad (2.11)$$

with a penalty parameter  $\eta_e$ , which was set to  $\eta_e = 1$  in the original formulation by Bassi and Rebay in [15] but is considered variable by Brezzi et al. in [26]. The calculation of the lifting operator based on (2.10) involves exact integration. However, from a computational viewpoint, using numerical integration simplifies the algorithms. Hereby, it is also reasonable to use the same the numerical quadrature rule which is already applied within the DG scheme. In case of a sufficient degree of exactness, i.e. for Legendre-Gauss quadrature, this essentially does not change the lifting operator. However, if the DG scheme is based on Legendre-Gauss-Lobatto integration, a second variant of the BR2 lifting operator is obtained due to the inexact numerical integration. We denote by  $\text{BR2}_{LGL}$  the variant based on Legendre-Gauss-Lobatto nodes while the lifting operator based on exact integration calculated e.g. by Legendre-Gauss quadrature is denoted by  $\text{BR2}_{LG}$ .

For the  $\text{BR2}_{LGL}$  flux, the lifting operator is therefore computed as

$$\langle l_j([u]), v \rangle_{LGL} = [u]_j \{v\}_j, \quad \forall v \in V_h, \quad (2.12)$$

where  $\langle \cdot, \cdot \rangle_{LGL}$  denotes the discrete inner product on  $V_h$  computed via Legendre-Gauss-Lobatto quadrature, i.e.

$$\langle v, w \rangle_{LGL} = \frac{\Delta x}{2} \sum_{j=1}^E \sum_{\nu=1}^{N+1} \omega_\nu u(\Lambda_j(\xi_\nu)) v(\Lambda_j(\xi_\nu)), \quad (2.13)$$

with the Legendre-Gauss-Lobatto quadrature nodes  $\xi_\nu$ ,  $\nu = 1, \dots, N+1$ , which are transformed to the given cell  $I_j$  by the map  $\Lambda_j$  defined in (1.43) and the corresponding quadrature weights  $\omega_\nu$ ,  $\nu = 1, \dots, N+1$ .

Considering the penalty parameter  $\eta_e$ , it has been shown by Brezzi et al. [26] using a coercivity condition, that the BR2 scheme is stable on triangular grids if  $\eta_e > 3$ . Since  $\eta_e$  is determined by the number of adjacent cells, this corresponds to  $\eta_e > 2$  for the one-dimensional case. In [162], Quaegebeur et al. obtained a sharper bound on  $\eta_e$  via energy stability considerations which yields  $\eta_e \geq \frac{N}{N+1}$  for the variant  $\text{BR2}_{LG}$ . This is the assertion of Theorem 2.4 in Section 2.3. However, the variant  $\text{BR2}_{LGL}$  is not considered in the analysis by Quaegebeur et

al. in [162]. Therefore, an extension of their results to the calculation of the lifting operator via Legendre-Gauss-Lobatto quadrature is provided by Theorem 2.6. The interesting result is that the BR2<sub>LGL</sub> scheme with parameter  $\eta_e$  is equivalent to the BR2<sub>LG</sub> scheme with parameter  $\hat{\eta}_e = \frac{N}{N+1}\eta_e$ . In addition, for the DG scheme on Legendre-Gauss-Lobatto nodes, the BR1 scheme is proven equivalent to BR2<sub>LGL</sub> for  $\eta_e = 1$ .

### Rewriting LDG<sub>a</sub>, LDG<sub>b</sub> and BR1 as second-derivative upwind SBP operators

In the following, we will identify generalized upwind SBP properties of the second-derivative operators given by the previously introduced specific DG diffusion discretizations, analogously to the construction (1.39) in Section 1.1.

Using the notation introduced in Section 1.2.1, we may rewrite the above variational formulation (2.4), (2.5) into a strong DG formulation similar to (1.51), whereby a transfer to the reference interval  $[-1, 1]$  is carried out. Thus, we obtain the equations

$$\begin{aligned} \frac{\Delta x}{2} \frac{d\mathbf{u}^j}{dt} &= d\mathbf{D}\mathbf{q}^j - d\mathbf{M}^{-1}[(q^j - q^{*,j})\mathbf{L}]_{-1}^1, \\ \frac{\Delta x}{2} \mathbf{q}^j &= \mathbf{D}\mathbf{u}^j - \mathbf{M}^{-1}[(u^j - u^{*,j})\mathbf{L}]_{-1}^1, \end{aligned}$$

for nodal values  $\mathbf{u}^j$  and  $\mathbf{q}^j$  of the primary and auxiliary variable, respectively, on each cell  $I_j$ . The above notation with respect to the numerical flux function is to be understood as

$$q^{*,j}(-1) = q_j^*, \quad u^{*,j}(-1) = u_j^* \quad \text{and} \quad q^{*,j}(1) = q_{j+1}^*, \quad u^{*,j}(1) = u_{j+1}^*.$$

For the LDG<sub>a</sub>, the LDG<sub>b</sub> and the BR1 scheme, the numerical flux  $q^*$  can now be written as

$$q_j^* = \left( \frac{1}{2} + \theta_q \right) q^-(x_j) + \left( \frac{1}{2} - \theta_q \right) q^+(x_j),$$

with

$$\theta_q = \begin{cases} \frac{1}{2} & \text{for LDG}_a, \\ -\frac{1}{2} & \text{for LDG}_b, \\ 0 & \text{for BR1}. \end{cases}$$

Furthermore, we have

$$u_j^* = \left( \frac{1}{2} + \theta_u \right) u^-(x_j) + \left( \frac{1}{2} - \theta_u \right) u^+(x_j),$$

with

$$\theta_u = \begin{cases} -\frac{1}{2} & \text{for LDG}_a, \\ \frac{1}{2} & \text{for LDG}_b, \\ 0 & \text{for BR1}. \end{cases}$$

Following the derivation of a global upwind SBP formulation for the DG-discretized one-dimensional advection equation in Section 1.2.3, the LDG<sub>a</sub> scheme in terms of the global nodal DG representation given by  $\mathbf{u} = (\mathbf{u}^1, \dots, \mathbf{u}^K)^T$ ,  $\mathbf{q} = (\mathbf{q}^1, \dots, \mathbf{q}^K)^T$  then rewrites as

$$\frac{d\mathbf{u}}{dt} = d\mathbf{D}_{glob}^-\mathbf{q}, \quad \mathbf{q} = \mathbf{D}_{glob}^+\mathbf{u},$$

with  $\mathbf{D}_{glob}^- = \mathbf{D}_{glob}^-(\theta = \frac{1}{2})$ , i.e. for the LDG<sub>a</sub> scheme, we have

$$\text{LDG}_a : \quad \frac{d\mathbf{u}}{dt} = d\mathbf{D}_2^+ \mathbf{u} = d\mathbf{D}_{glob}^- \mathbf{D}_{glob}^+ \mathbf{u}.$$

Analogously, for the LDG<sub>b</sub> variant we obtain

$$\text{LDG}_b : \quad \frac{d\mathbf{u}}{dt} = d\mathbf{D}_2^- \mathbf{u} = d\mathbf{D}_{glob}^+ \mathbf{D}_{glob}^- \mathbf{u}.$$

Both of these formulations correspond to true upwind SBP operators approximating the second derivative as defined in (1.39). In contrast, for the BR1 scheme, we have

$$\text{BR1} : \quad \frac{d\mathbf{u}}{dt} = d\mathbf{D}_2 \mathbf{u} = d\mathbf{D}_{glob} \mathbf{D}_{glob} \mathbf{u},$$

where  $\mathbf{D}_{glob} = \mathbf{D}_{glob}^-(\theta = 0) = \mathbf{D}_{glob}^+(\theta = 0)$  is a generalized SBP first-derivative operator in the sense of Definition 1.3, as already mentioned in Remark 1.12. Thus, the BR1 scheme represents a second-derivative SBP operator which is obtained by applying the above first-derivative SBP operator twice, similar to the definition for finite difference schemes by (1.17).

## 2.2 The $(\sigma, \mu)$ -family of DG diffusion discretizations

The so-called  $(\sigma, \mu)$ -family of DG diffusion discretizations has been revisited more recently in a series of works by van Leer et al. in [111, 119, 109]. In fact, diffusion discretizations of this type have first been recognized as a two-parameter family in a PhD thesis by van Raalte [163] although specific members of this family were already well-known. Actually, the earliest approaches to discretize viscous fluxes within the DG framework can be cast into this formulation since this family naturally includes the original DG diffusion discretizations in [5, 17].

Various more recent DG diffusion approaches also lie within the  $(\sigma, \mu)$ -family. More precisely, if the integrals in the DG variational form are numerically solved by Legendre-Gauss-Lobatto quadrature, the  $(\sigma, \mu)$ -family contains both the BR1 and BR2 scheme as well as a symmetrized form of LDG, see [109]. Moreover, in [111, 119], the  $(\sigma, \mu)$ -family is considered a starting point for the recovery approach to the discretization of diffusion fluxes in the context of DG schemes.

Now, the element-wise variational formulation of the  $(\sigma, \mu)$ -family of DG diffusion discretizations reads

$$(u_t, v)_j = d\mathcal{L}_j(u, v), \quad \forall v \in V_h, \quad \forall j = 1, \dots, E, \quad (2.14)$$

where the operator  $\mathcal{L}_j(\cdot, \cdot)$  referring to an element  $I_j$  is given by

$$\begin{aligned} \mathcal{L}_j(u, v) &= -(u_x, v_x)_j + \{u_x\}_{j+1} v_{j+1}^- - \{u_x\}_j v_j^+ \\ &\quad + \frac{\sigma}{2} \left( (v_x^-[u])_{j+1} + (v_x^+[u])_j \right) + \frac{\mu}{\Delta x} \left( ([u]v^-)_{j+1} - ([u]v^+)_j \right). \end{aligned} \quad (2.15)$$

This DG diffusion discretization is hence specified by the two parameters  $\sigma, \mu \in \mathbb{R}$ . Hereby, the choice  $\sigma = -1, \mu \geq 1$  corresponds to the interior penalty scheme of Arnold [5] which is symmetric and stable, while  $\sigma = 1, \mu = 0$  yields the non-symmetric but stable scheme of Baumann and Oden [17], see also the revision of this family of diffusion discretizations in [109].

### 2.2.1 Connection to contemporary DG diffusion discretizations

Considering the nodal DG scheme on Legendre-Gauss-Lobatto nodes using mass lumping, one may cast the BR1 scheme [13] and the BR2 scheme [14] as well as a symmetric variant of the LDG scheme [44] into the framework of the  $(\sigma, \mu)$ -family, as will be shown in the following.

For this purpose, we set  $r = v_x$  in order to insert (2.5) into (2.4), yielding

$$(u_t, v)_j = d \left( (u, v_{xx})_j - (u^* v_x^-)_{j+1} + (u^* v_x^+)_{j+1} + q_{j+1}^* v_{j+1}^- - q_j^* v_j^+ \right).$$

Using partial integration  $(u, v_{xx})_j = (uv_x)_j^- - (uv_x)_j^+ - (u_x, v_x)_j$ , the above equation now rewrites as

$$(u_t, v)_j = d \left( -(u_x, v_x)_j - ((u^* - u^-)v_x^-)_{j+1} + ((u^* - u^+)v_x^+)_{j+1} \right) + d \left( q_{j+1}^* v_{j+1}^- - q_j^* v_j^+ \right). \quad (2.16)$$

Furthermore, partially integrating the term  $-(u, r_x)$  in (2.5), we have

$$(q, r)_j = (u_x, r)_j + (u^* - u^-)_{j+1} r_{j+1}^- - (u^* - u^+)_{j+1} r_j^+. \quad (2.17)$$

Although the linear heat equation enables an exact evaluation of the integrals occurring in the DG formulation, spatial integration is usually carried out numerically in order to allow for extensions also to nonlinear equations. Equivalence of specific  $(\sigma, \mu)$ -schemes to contemporary DG schemes for diffusion necessitates that numerical integration is carried out using the Legendre-Gauss-Lobatto quadrature rule, which computes the left-hand side terms  $(u_t, v)_j$  and  $(q, r)_j$  in (2.16) and (2.17), respectively, with reduced accuracy.

Connections to contemporary DG schemes will be established in the following paragraphs by inserting suitable test functions  $r$  into (2.17). For this purpose, global functions on the complete domain  $\Omega = \cup_{j=1}^E I_j$  will be defined from local ones on cells  $I_j$ . Therefore, we need to define the surjection  $\mathcal{M}_j$  for a given cell  $I_j$  by

$$\mathcal{M}_j : \Omega \rightarrow [-1, 1], \quad x \mapsto \frac{2x - (x_j + x_{j+1})}{\Delta x} \chi|_{I_j},$$

using the indicator function  $\chi|_{I_j}$  with  $\chi|_{I_j}(x) = 1$  if  $x \in I_j$  and  $\chi|_{I_j}(x) = 0$ .

#### The BR1 scheme

In order to obtain the representation of BR1 as a member of the  $(\sigma, \mu)$ -family, we insert the Lagrange polynomials  $L_1, L_{N+1}$  corresponding to the first and last Legendre-Gauss-Lobatto node, i.e. to the boundary nodes  $\xi_1 = -1$  and  $\xi_{N+1} = 1$ , as test functions into (2.17). As stated before, the integrals in (2.17) are numerically computed using the Legendre-Gauss-Lobatto quadrature rule. Setting  $r = L_1 \circ \mathcal{M}_j \chi|_{I_j}$  in (2.17) with  $r_j^+ = 1$ ,  $r_{j+1}^- = 0$ , and keeping in mind  $u_j^{*,BR1} = \{u\}_j$ , we hence obtain

$$q_j^+ = (u_x)_j^+ + \frac{1}{\Delta x \omega_1} [u]_j. \quad (2.18)$$

On the other hand, setting  $r = L_{N+1} \circ \mathcal{M}_j \chi|_{I_j}$  in (2.17) with  $r_j^+ = 0$ ,  $r_{j+1}^- = 1$ , and using symmetry of the weights  $\omega_{N+1} = \omega_1$ , we have

$$q_{j+1}^- = (u_x)_{j+1}^- + \frac{1}{\Delta x \omega_1} [u]_{j+1}. \quad (2.19)$$

Inserting the BR1 fluxes  $u_j^{*,BR1} = \{u\}_j$  and  $q_j^{*,BR1} = \{u_x\}_j + \frac{1}{\Delta x \omega_1} [u]_j$  into (2.16), we then have

$$\begin{aligned} (u_t, v)_j &= d \left( -(u_x, v_x)_j - \frac{1}{2} ([u]v_x^-)_{j+1} - \frac{1}{2} ([u]v_x^+)_j \right) \\ &\quad + d \left( \{u_x\}_{j+1} v_{j+1}^- + \frac{1}{\Delta x \omega_1} [u]_{j+1} v_{j+1}^- \right) \\ &\quad - d \left( \{u_x\}_j v_j^+ + \frac{1}{\Delta x \omega_1} [u]_j v_j^+ \right). \end{aligned}$$

Comparing with the definition of the operator  $\mathcal{L}_j$  in (2.15), the parameters of the BR1 scheme are thus  $\sigma = -1$  and  $\mu = \frac{1}{\omega_1}$ .

### A symmetric form of the LDG scheme

Now, we consider the arithmetic average of the two alternating variants of the LDG scheme. For the LDG<sub>a</sub> fluxes with  $u^{*,LDG_a} = u^+$  and  $u^{*,LDG_a} = q^-$ , inserting the test function  $r = L_{N+1} \circ \mathcal{M}_j \chi|_{I_j}$  into (2.17), and shifting the right-hand side index  $j+1$  to  $j$ , we obtain

$$q_j^{*,LDG_a} = q_j^- = (u_x)_j^- + \frac{2}{\Delta x \omega_1} [u]_j.$$

Analogously, for the LDG<sub>b</sub> flux with  $u^{*,LDG_b} = u^-$  and  $q^{*,LDG_b} = q^+$ , inserting the test function  $r = L_1 \circ \mathcal{M}_j \chi|_{I_j}$  into (2.17) yields

$$q_j^{*,LDG_b} = q_j^+ = (u_x)_j^+ + \frac{2}{\Delta x \omega_1} [u]_j.$$

The arithmetic average of the two alternating variants of the LDG fluxes is then given by inserting the arithmetic means

$$u_j^* = \frac{1}{2} (u_j^{*,LDG_a} + u_j^{*,LDG_b}) = \{u\}_j, \quad q_j^* = \frac{1}{2} (q_j^{*,LDG_a} + q_j^{*,LDG_b}) = \{u_x\}_j + \frac{2}{\Delta x \omega_1} [u]_j$$

into (2.16). Thus, we have

$$\begin{aligned} (u_t, v)_j &= d \left( -(u_x, v_x)_j - \frac{1}{2} ([u]v_x^-)_{j+1} - \frac{1}{2} ([u]v_x^+)_j \right) \\ &\quad + d \left( \{u_x\}_{j+1} v_{j+1}^- + \frac{2}{\Delta x \omega_1} [u]_{j+1} v_{j+1}^- \right) \\ &\quad - d \left( \{u_x\}_j v_j^+ + \frac{2}{\Delta x \omega_1} [u]_j v_j^+ \right). \end{aligned}$$

The parameters of this symmetric variant of the LDG scheme are thus  $\sigma = -1$  and  $\mu = \frac{2}{\omega_1}$ .

### The BR2 scheme

For the BR2 approach, the numerical flux  $u^*$  used in the auxiliary equation is again given by  $u_j^{*,BR2} = \{u\}_j$  and the numerical flux  $q_j^{*,BR2}$  is determined by the local lifting operators  $l_j$  calculated either as an exact projection of the jump  $[u]_j$  in (2.10) or by the equation (2.12) based on Legendre-Gauss-Lobatto quadrature. In the later case, inserting the test functions  $v_1 = L_1 \circ \mathcal{M}_j \chi|_{I_j}$  and  $v_2 = L_{N+1} \circ \mathcal{M}_{j-1} \chi|_{I_{j-1}}$  into (2.12) results in

$$\{l_j([u])\}_j = \frac{1}{\Delta x \omega_1} [u]_j$$

and thus

$$q_j^{*,BR2} = \{u_x\}_j + \frac{\eta_e}{\Delta x \omega_1} [u]_j.$$

Analogously to the derivation for the BR1 scheme and the symmetrized arithmetic mean LDG scheme in the previous paragraphs, inserting the fluxes  $u^{*,BR2}$  and  $q^{*,BR2}$  into (2.16) now identifies the BR2 scheme as a  $(\sigma, \mu)$ -scheme with  $\sigma = -1$  and  $\mu = \frac{\eta_e}{\omega_1}$ .

### The $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme

An interesting member of the  $(\sigma, \mu)$ -family is the  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme [111, 109]. For a polynomial degree of  $N = 1$ , it can be interpreted by a recovery approach where a continuous representation of the DG approximation over two adjacent cells is constructed and used to approximate the diffusion flux. More precisely, for each interface, the piece-wise linear DG representation on the two adjacent cells is replaced by a single cubic function on the union of those two cells such that the piece-wise linear DG approximation is the  $L^2$ -projection of the recovered cubic function. By using a cubic representation, recovery is done with maximum accuracy. In fact, the  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme is formally forth order accurate.

#### 2.2.2 An energy estimate for the global diffusion operator

Next, we provide the following useful properties regarding dissipativity of the  $(\sigma, \mu)$ -family under specific conditions on  $\sigma$  and  $\mu$  on the one hand, and boundedness of the interface terms of the formulation on the other hand. In particular, these properties will be needed in Section 3.2 for the  $L^2$ -stability analysis of specific DG methods coupled with IMEX time integration schemes based on advection-diffusion IMEX splitting.

Built from the local diffusion operators  $\mathcal{L}_j$ , we define the corresponding global operator of a given  $(\sigma, \mu)$ -scheme by

$$\mathcal{L}(u, v) = \sum_{j=1}^E \mathcal{L}_j(u, v).$$

For this global operator, we have

$$\mathcal{L}(u, v) = -(u_x, v_x) - \sum_{j=1}^E \{u_x\}_j [v]_j + \sigma \sum_{j=1}^E \{v_x\}_j [u]_j - \frac{\mu}{\Delta x} \sum_{j=1}^E [u]_j [v]_j. \quad (2.20)$$

In order to obtain an energy estimate regarding the diffusion operator, we will need to provide an upper bound for the interface terms in the above definition. Based on the classical  $L^2$  inner product  $(\cdot, \cdot)$  the corresponding  $L^2$ -norm  $\|\cdot\|$  is defined on  $V_h$ , i.e.

$$\|v\|^2 = (v, v) = \sum_{j=1}^E (v, v)_j.$$

In addition, we define the jump semi-norm  $[[v]]$  for  $v \in V_h$  via

$$[[v]]^2 = \sum_{j=1}^E [v]_j^2.$$

We now have the following auxiliary estimate for the mixed interface terms occurring in the definition of the diffusion operator  $\mathcal{L}$ .

**Lemma 2.1.** *For any  $u, v \in V_h$  and for any constant  $\tilde{C} > 0$ , a bound on the boundary terms occurring in the definition of  $\mathcal{L}(u, v)$  is given by*

$$\left| \sum_{j=1}^E \{u_x\}_j [v]_j \right| \leq \tilde{C} \|u_x\|^2 + \frac{1}{4\tilde{C}\Delta x \omega_1} [[v]]^2. \quad (2.21)$$

*Proof.* Since this estimate is trivial for the first-order DG scheme as in this case the piece-wise derivative vanishes on each cell, we only consider higher than first-order schemes, i.e.  $N \geq 1$  in the definition (2.2) of  $V_h$ .

We note that the piece-wise derivative  $u_x$  of  $u \in V_h$  fulfills  $u_x|_{I_j} \in P^{N-1}(I_j)$ . Therefore, Legendre-Gauss-Lobatto (LGL) integration exactly computes  $\|u_x\|^2$ . To incorporate the exact numerical integration by LGL quadrature into our analysis, we reuse the notation in (2.13) for this quadrature rule which has been introduced in Section 2.1 and incorporates the transformation of  $I_j$  to the reference interval  $[-1, 1]$  by the map  $\Lambda_j$ . For the norm  $\|u_x\|$ , we then have

$$\begin{aligned} \|u_x\|^2 &= \sum_{j=1}^E \|u_x\|_j^2 = \langle u_x, u_x \rangle_{LGL} = \frac{\Delta x}{2} \sum_{\substack{1 \leq j \leq E \\ 1 \leq \nu \leq N+1}} \omega_\nu(u_x(\Lambda_j(\xi_\nu)))^2 \\ &= \frac{\Delta x}{2} \left( \sum_{\substack{1 \leq j \leq E \\ 1 < \nu < N+1}} \omega_\nu(u_x(\Lambda_j(\xi_\nu)))^2 + \sum_{j=1}^E \omega_1(u_x(\Lambda_j(\xi_1)))^2 + \sum_{j=1}^E \omega_{N+1}(u_x(\Lambda_j(\xi_{N+1})))^2 \right) \\ &= \frac{\Delta x}{2} \sum_{\substack{1 \leq j \leq E \\ 1 < \nu < N+1}} \omega_\nu(u_x(\Lambda_j(\xi_\nu)))^2 + \Delta x \sum_{j=1}^E \omega_1 \{u_x^2\}_j, \quad \text{as } \omega_{N+1} = \omega_1. \end{aligned}$$

Hence, we may estimate the term  $\sum_j \{u_x\}_j [v]_j$  by

$$\begin{aligned} \left| \sum_{j=1}^E \{u_x\}_j [v]_j \right| &\leq \frac{1}{2} \sum_{j=1}^E |u_x(\Lambda_j(\xi_1)) [v]_j| + |u_x(\Lambda_{j-1}(\xi_{N+1})) [v]_j| \\ &\leq \frac{1}{2} \sum_{j=1}^E \left( \tilde{C} \Delta x \omega_1 (u_x^2(\Lambda_j(\xi_1)) + u_x^2(\Lambda_{j-1}(\xi_{N+1}))) + \frac{1}{2\tilde{C}\Delta x \omega_1} [v]_j^2 \right) \\ &\leq \tilde{C} \langle u_x, u_x \rangle_{LGL} + \frac{1}{4\tilde{C}\Delta x \omega_1} \sum_{j=1}^E [v]_j^2 \\ &= \tilde{C} \|u_x\|^2 + \frac{1}{4\tilde{C}\Delta x \omega_1} [[v]]^2, \end{aligned}$$

for any constant  $\tilde{C} > 0$ .  $\square$

Using the above intermediate result, we can prove the following Theorem regarding dissipativity of the discrete diffusion operator  $\mathcal{L}$ .

**Theorem 2.2.** *The diffusion operator  $\mathcal{L}$  fulfills the following properties.*

1. *If the parameters  $\sigma$  and  $\mu$  fulfill the condition  $\frac{(1-\sigma)^2}{4\omega_1} \leq \mu$ , we have dissipativity of  $\mathcal{L}$ , i.e.  $\mathcal{L}(u, u) \leq 0$  for any  $u \in V_h$ .*
2. *Let  $\mathbf{u}, \mathbf{v} \in (V_h)^n$ , and define a vectorized version of the diffusion operator as*

$$\underline{\mathcal{L}}(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^n \mathcal{L}(u_i, v_i).$$

*If again  $\frac{(1-\sigma)^2}{4\omega_1} \leq \mu$  is fulfilled, then we have the following assertions.*

- (a) *For  $\mathbf{u} \in (V_h)^n$ , we have  $\underline{\mathcal{L}}(\mathbf{u}, \mathbf{u}) \leq 0$ .*
- (b) *For  $A \in \mathbb{R}^{n \times n}$  symmetric, we have*

$$\underline{\mathcal{L}}(\mathbf{u}, A\mathbf{v}) = \underline{\mathcal{L}}(A\mathbf{u}, \mathbf{v}).$$

- (c) *For  $B \in \mathbb{R}^{n \times n}$  symmetric and positive definite, we have*

$$\underline{\mathcal{L}}(\mathbf{u}, B\mathbf{v}) \leq 0.$$

*Proof.* 1. This assertion follows from summing up the cell-wise definition (2.15) of the diffusion operator. We have

$$\mathcal{L}(u, u) = -(u_x, u_x) - (1 - \sigma) \sum_j \{u_x\}_j [u]_j - \frac{\mu}{\Delta x} [[u]]^2.$$

If  $\sigma = 1$ , we directly obtain  $\mathcal{L}(u, u) \leq 0$ . Otherwise, setting  $\tilde{C} = \frac{1}{|1-\sigma|}$  in the inequality (2.21) given in the previous Lemma 2.1 yields

$$\mathcal{L}(u, u) \leq \left( \frac{(1-\sigma)^2}{4\Delta x \omega_1} - \frac{\mu}{\Delta x} \right) [[u]]^2.$$

Therefore, choosing the parameters of the  $(\sigma, \mu)$ -family such that they fulfill  $\frac{(1-\sigma)^2}{4\omega_1} \leq \mu$ , we have the dissipative property  $\mathcal{L}(u, u) \leq 0$ .

2. (a) This assertion obviously follows from the definition of  $\underline{\mathcal{L}}$  since

$$\underline{\mathcal{L}}(\mathbf{u}, \mathbf{u}) = \sum_{i=1}^n \mathcal{L}(u_i, u_i) \leq 0.$$

- (b) Since  $\underline{\mathcal{L}}$  inherits bi-linearity from  $\mathcal{L}$ , a simple calculation using the symmetry of  $A$  yields

$$\begin{aligned} \underline{\mathcal{L}}(\mathbf{u}, A\mathbf{v}) &= \sum_{i=1}^n \sum_{k=1}^n a_{ik} \mathcal{L}(u_i, u_k) = \sum_{k=1}^n \mathcal{L}\left(\sum_{i=1}^n a_{ik} u_i, u_k\right) \\ &= \underline{\mathcal{L}}(A\mathbf{u}, \mathbf{v}). \end{aligned}$$

- (c) Since  $B$  is symmetric and positive definite, it has a unique symmetric square root  $A$ . Using the previous assertions, we thus obtain

$$\underline{\mathcal{L}}(\mathbf{u}, B\mathbf{u}) = \underline{\mathcal{L}}(\mathbf{u}, A^2\mathbf{u}) = \underline{\mathcal{L}}(A\mathbf{u}, A\mathbf{u}) \leq 0.$$

□

## 2.3 Energy stability of the BR2 scheme

Recently, Quaegebeur et al. [162] investigated the class of ESFR schemes described in Section 1.3 in terms of energy stability for the one-dimensional linear diffusion equation using various compact numerical diffusion fluxes. As already stated in Section 1.3, the class of ESFR schemes is based on the flux reconstruction framework developed by Huynh [85] which has also been applied to diffusion equations in [86]. The original identification of ESFR schemes by Vincent et al. in [202] aimed at energy stability for the linear advection equation. Castonguay et al. [37] extended this stability proof to discretization of the linear diffusion equation by ESFR schemes employing general LDG numerical diffusion fluxes of the form (2.7) with non-negative penalty parameter  $c_{11}$ . The results by Quaegebeur et al. [162] thus represent an extension of the theoretical analysis in [37] to more general classes of compact numerical diffusion fluxes.

In this section, we will further extend the results by Quaegebeur et al. regarding energy stability of the BR2 scheme with respect to the penalty parameter  $\eta_e$  by the variant of calculating the BR2 lifting operator via Legendre-Gauss-Lobatto quadrature. In addition, we obtain a

simplified representation of the lower bound  $\eta_e = \frac{N}{N+1}$  for energy stability of the BR2 scheme in case of the calculation of the BR2 lifting operator by exact projection.

Considering the first order reformulation of the linear diffusion equation given in (2.3) and the approximation space  $V_h$  defined in (2.2), the ESFR schemes construct a piecewise polynomial solution  $u(t) \in V_h$  approximating  $U(x, t)$  and an auxiliary quantity  $q(t) \in V_h$  approximating the auxiliary variable  $Q(x, t)$ , similarly to DG schemes. As described in Section 1.3, the ESFR schemes are characterized by the chosen correction functions to deal with the discontinuous approximation space. For the first-order system (2.3), the correction functions are denoted by  $h_L, h_R : [-1, 1] \rightarrow \mathbb{R}$  for the primary equation and by  $g_L, g_R : [-1, 1] \rightarrow \mathbb{R}$  for the auxiliary equation, respectively.

Analogously to (1.91), we obtain the ESFR schemes for the linear diffusion equation on general non-overlapping one-dimensional grids by the cell-wise formulation

$$\frac{\partial u_j}{\partial t}(\xi, t) = d \frac{2}{\Delta x_j} \left[ \frac{\partial q_j}{\partial \xi}(\xi, t) + (q_j^* - q_j(-1, t)) h'_L(\xi) + (q_{j+1}^* - q_j(1, t)) h'_R(\xi) \right], \quad (2.22)$$

$$q_j = \frac{2}{\Delta x_j} \left[ \frac{\partial u_j}{\partial \xi}(\xi, t) + (u_j^* - u_j(-1, t)) g'_L(\xi) + (u_{j+1}^* - u_j(1, t)) g'_R(\xi) \right], \quad (2.23)$$

with not necessarily uniform cell lengths  $\Delta x_j$ . Hereby, we again denote the representation of the approximate solution on each cell  $I_j$  by

$$u_j(\xi, t) = u(\Lambda_j(\xi), t), \quad q_j(\xi, t) = q(\Lambda_j(\xi), t),$$

with the reference coordinate  $\xi \in [-1, 1]$  and the mapping  $\Lambda_j$  to a specific cell  $I_j$  defined in (1.43). Numerical diffusion fluxes are now specified by a suitable choice of  $q_j^*$  and  $u_j^*$ . Furthermore, the correction functions derived in [202] in order to obtain energy stability have the form

$$g_{L,\kappa} = \frac{(-1)^N}{2} \left[ \Psi_N - \frac{\eta_{N,\kappa} \Psi_{N-1} + \Psi_{N+1}}{1 + \eta_{N,\kappa}} \right], \quad g_{R,\kappa}(\xi) = g_{L,\kappa}(-\xi), \quad (2.24)$$

$$h_{L,c} = \frac{(-1)^N}{2} \left[ \Psi_N - \frac{\eta_{N,c} \Psi_{N-1} + \Psi_{N+1}}{1 + \eta_{N,c}} \right], \quad h_{R,c}(\xi) = h_{L,c}(-\xi), \quad (2.25)$$

where  $\Psi_N$  denotes the Legendre polynomial of degree  $N$  and the parameters are given by

$$\eta_{N,\kappa} = \frac{\kappa(2N+1)(a_N N!)^2}{2}, \quad \eta_{N,c} = \frac{c(2N+1)(a_N N!)^2}{2},$$

with  $a_N = \frac{(2N)!}{2^N (N!)^2}$  depending on the polynomial degree.

The correction polynomials defined in (2.24) and (2.25) thus depend on the two parameters  $\kappa$  and  $c$ . Therefore, the primary and auxiliary equation are not necessarily discretized by the same member of the class of ESFR schemes. Considering the representation of nodal DG schemes in ESFR framework, the DG scheme on Legendre-Gauss nodes chooses the left and right Legendre-Gauss-Radau polynomials as correction polynomials for both equations, i.e.  $\eta_{N,\kappa} = \eta_{N,c} = 0$ , or else  $\kappa = c = 0$ , while the DG scheme on Legendre-Gauss-Lobatto nodes is obtained by choosing  $\eta_{N,\kappa} = \eta_{N,c} = \frac{N+1}{N}$ , see e.g. Huynh [85].

In particular, Quaeghebeur et al. [162] showed that for ESFR schemes, the  $\text{BR2}_{LG}$  diffusion fluxes with exact projection property to calculate the lifting operators  $l_j$  are equivalent to the interior penalty (IP) fluxes

$$u^{*,IP} = \{u\}, \quad q^{*,IP} = \{u_x\} + \tau[u] \quad (2.26)$$

if the penalty parameter  $\tau \geq 0$  is chosen appropriately. Therefore, energy stability of the ESFR schemes with IP diffusion fluxes carries over to ESFR schemes with  $\text{BR2}_{LG}$  diffusion fluxes.

More precisely, from [162], we have the two following results for ESFR schemes constructing a piecewise polynomial solution of degree  $N$  on a non-overlapping grid in one space dimension.

**Theorem 2.3.** *Employing IP fluxes within ESFR schemes for the linear diffusion equation is energy stable if the penalty parameter  $\tau$  in (2.26) is chosen such that  $\tau_j \geq \tau_j^*$  holds locally on each cell interface, with*

$$\tau_j^* = \frac{1}{2} \left( \frac{\Delta x_{j-1} + \Delta x_j}{\Delta x_{j-1} \Delta x_j} \right) \min_{\kappa} (|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1)) . \quad (2.27)$$

**Theorem 2.4.** *The  $\text{BR2}_{LG}$  formulation is equivalent to the IP formulation if and only if*

$$\tau = \frac{\Delta x_{j-1} + \Delta x_j}{4 \Delta x_{j-1} \Delta x_j} (N+1)^2 \eta_e . \quad (2.28)$$

*Furthermore, employing  $\text{BR2}_{LG}$  fluxes within ESFR schemes for the linear diffusion equation is energy stable if the penalty parameter  $\eta_e$  in (2.11) is chosen such that  $\eta_e \geq \eta_e^*$  with*

$$\eta_e^* = \frac{2 \min_{\kappa} (|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1))}{(N+1)^2} . \quad (2.29)$$

The above relations by Quaeghebeur et al. may be simplified using the following auxiliary result based on the properties of the Legendre polynomials.

**Lemma 2.5.** *For the parameter-dependent ESFR correction polynomial  $g_{L,\kappa}$  of degree  $N+1$  defined in (2.24), we have*

$$\min_{\kappa} (|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1)) = \frac{N(N+1)}{2} . \quad (2.30)$$

*Proof.* The Legendre polynomials occurring in the definition of the correction polynomials as well as their first derivatives have well-known boundary values. Specifically, e.g. from [1], we have

$$\Psi'_N(1) = \frac{N(N+1)}{2} \text{ and } \Psi'_N(-1) = (-1)^{N+1} \frac{N(N+1)}{2} .$$

Inserting this into (2.24), we have

$$\begin{aligned} 4 (|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1)) &= \left| N(N+1) - \frac{\eta_{N,\kappa} N(N-1) + (N+1)(N+2)}{1 + \eta_{N,\kappa}} \right| \\ &\quad + N(N+1) + \frac{\eta_{N,\kappa} N(N-1) + (N+1)(N+2)}{1 + \eta_{N,\kappa}} . \end{aligned} \quad (2.31)$$

If  $N(N+1) < \frac{\eta_{N,\kappa} N(N-1)+(N+1)(N+2)}{1+\eta_{N,\kappa}}$ , e.g. for the DG scheme on Legendre-Gauss nodes with  $\eta_{N,\kappa} = 0$ , we have

$$|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1) = \frac{1}{2} \frac{\eta_{N,\kappa} N(N-1) + (N+1)(N+2)}{1+\eta_{N,\kappa}} > \frac{N(N+1)}{2}.$$

Therefore, the minimum value of the above term is obviously attained for values of  $\kappa$  such that  $N(N+1) \geq \frac{\eta_{N,\kappa} N(N-1)+(N+1)(N+2)}{1+\eta_{N,\kappa}}$ , e.g. for the DG scheme on Legendre-Gauss-Lobatto nodes with  $\eta_{N,\kappa} = \frac{N+1}{N}$ . For these values of  $\eta_{N,\kappa}$ , the minimum value of

$$|g'_{L,\kappa}(1)| - g'_{L,\kappa}(-1) = \frac{N(N+1)}{2}$$

is obtained, which proves the assertion.  $\square$

Inserting the above equality (2.30) into (2.27) and (2.29) yields a simplification of the parameter restrictions of the form

$$\tau_j^* = \frac{\Delta x_{j-1} + \Delta x_j}{4\Delta x_{j-1}\Delta x_j} N(N+1), \quad (2.32)$$

$$\eta_e^* = \frac{N}{N+1}. \quad (2.33)$$

For the IP formulation on uniform grids with equal cell lengths, this further simplifies to

$$\tau_j^* = \tau^* = \frac{N(N+1)}{2\Delta x}.$$

Regarding the computation of the BR2 lifting operator by Legendre-Gauss-Lobatto quadrature, an extension of the analysis by Quaegebeur et al. yields the following result for the BR2<sub>LGL</sub> diffusion fluxes employed within ESFR schemes of degree  $N$  on non-overlapping one-dimensional grids.

**Theorem 2.6.** *The BR2<sub>LGL</sub> formulation with parameter  $\eta_e$  is equivalent to the BR2<sub>LG</sub> formulation with parameter  $\hat{\eta}_e = \frac{N}{N+1}\eta_e$ . Therefore, employing BR2<sub>LGL</sub> fluxes within ESFR schemes for the linear diffusion equation is energy stable if  $\eta_e \geq 1$ .*

*Proof.* In case of Legendre-Gauss-Lobatto quadrature, the cell boundaries are contained in the nodal set. Hence, we have

$$\{l_j([u])\}_j = \frac{1}{\omega_{N+1}\Delta x_{j-1}} \langle l_j([u]), L_{N+1} \circ \mathcal{M}_{j-1} \chi|_{I_{j-1}} \rangle_{LGL} + \frac{1}{\omega_1\Delta x_j} \langle l_j([u]), L_1 \circ \mathcal{M}_j \chi|_{I_j} \rangle_{LGL},$$

where  $L_1$  and  $L_{N+1}$  are the Lagrange polynomials corresponding to the boundary nodes  $\xi_1 = -1$  and  $\xi_{N+1} = 1$  and  $\omega_1 = \omega_{N+1} = \frac{2}{N(N+1)}$  are the corresponding Legendre-Gauss-Lobatto quadrature weights, see e.g. [1]. Inserting the weights and further simplification yields

$$\{l_j([u])\}_j = \frac{N(N+1)}{4} [u]_j \left( \frac{1}{\Delta x_{j-1}} + \frac{1}{\Delta x_j} \right) = N(N+1) \frac{\Delta x_{j-1} + \Delta x_j}{4\Delta x_{j-1}\Delta x_j} [u]_j. \quad (2.34)$$

Hence, for ESFR schemes, the BR2<sub>LGL</sub> diffusion discretization is equivalent to an IP formulation with  $\tau_j = \eta_e N(N+1) \frac{\Delta x_{j-1} + \Delta x_j}{4\Delta x_{j-1}\Delta x_j}$ . Comparing with (2.28), the BR2<sub>LGL</sub> formulation is therefore also equivalent to the BR2<sub>LG</sub> formulation with penalty parameter  $\hat{\eta}_e = \frac{N}{N+1}\eta_e$ .

According to Theorem 2.3 with  $\tau_j^*$  in the simplified form (2.32), employing the BR2<sub>LGL</sub> diffusion fluxes within an ESFR scheme thus results in an energy stable scheme if  $\eta_e \geq 1$ .  $\square$

Furthermore, for DG schemes on Legendre-Gauss-Lobatto nodes, the equivalence result of Theorem 2.6 is supplemented by equivalence to the BR1 formulation. More precisely, for this particular ESFR scheme, the BR1 formulation is recovered by a particular choice of BR2 as shown in the following

**Theorem 2.7.** *For the DG scheme on Legendre-Gauss-Lobatto nodes, the BR1 scheme, the BR2<sub>LGL</sub> scheme with  $\eta_e = 1$ , and the BR2<sub>LG</sub> scheme with  $\eta_e = \frac{N}{N+1}$  are equivalent.*

*Proof.* Obviously, due to the assertion of equivalence in Theorem 2.6, proving the equivalence of the BR1 scheme to the BR2<sub>LGL</sub> scheme with  $\eta_e = 1$  is sufficient.

The BR1 formulation within a nodal DG scheme using Legendre-Gauss-Lobatto quadrature has already been identified as a member of the  $(\sigma, \mu)$ -family in Section 2.2.1. In this context, the left-hand side and right-hand side values of the auxiliary quantity  $q$  in a specific cell  $I_j$  are derived from the auxiliary equation (2.17). On general non-overlapping grids, the uniform cell length  $\Delta x$  appearing in equations (2.18) and (2.19) is replaced by the length  $\Delta x_j$  and the precise value of the first Legendre-Gauss-Lobatto weight  $\omega_1 = \frac{2}{N(N+1)}$  is inserted. Thus, we obtain

$$\begin{aligned} q_j^+ &= (u_x)_j^+ + \frac{N(N+1)}{2\Delta x_j} [u]_j, \\ q_{j+1}^- &= (u_x)_{j+1}^- + \frac{N(N+1)}{2\Delta x_j} [u]_{j+1}. \end{aligned}$$

Hence, the diffusion flux for the auxiliary variable  $q$  at the cell interface  $x_j$  between the cells  $I_{j-1}$  and  $I_j$  is given by

$$q_j^{*,BR1} = \{q\}_j = \frac{1}{2} (q_j^- + q_j^+) = \{u_x\}_j + N(N+1) \frac{\Delta x_{j-1} + \Delta x_j}{4\Delta x_{j-1}\Delta x_j} [u]_j = q_j^{*,BR2_{LGL}(\eta_e=1)}.$$

$\square$

## 2.4 A comparative Fourier analysis for linear advection-diffusion problems

In this section, we review the investigation of the wave propagation properties of discontinuous Galerkin schemes for advection-diffusion problems with respect to several classical discretizations of the diffusion terms which was carried out by the author in [148]. Specifically, the influence on dissipation and dispersion properties of the two alternating versions of the LDG scheme as well as the BR1 and the BR2 scheme is studied in [148]. The analysis highlights

a significant difference between the two possible ways to choose the alternating LDG fluxes showing that the variant which is inconsistent with the upwind advective flux is more accurate in case of advection-diffusion discretizations. Furthermore, while for the BR1 scheme used within a third order DG scheme on Legendre-Gauss nodes, a higher accuracy for well-resolved problems has previously been observed in the literature, we will see that higher accuracy of the BR1 discretization only holds for odd orders of the DG scheme. In addition, this higher accuracy is generally lost on Legendre-Gauss-Lobatto nodes.

The investigation of wave propagation properties in terms of dispersion and diffusion errors depending on the wave number is of utmost importance for the analysis of accuracy and stability of any numerical scheme applied in the context of computational fluid dynamics. Especially in case of high order methods and for under-resolved turbulence simulation, a desired small numerical dissipation competes with robustness and thus has to be carefully analyzed. Therefore, dispersion and diffusion properties have been investigated for major classes of high order schemes such as the DG scheme [82], the spectral difference method [196], flux reconstruction schemes [203] and continuous Galerkin (CG) approximations [138].

Dissipation and dispersion properties are usually inspected via Fourier analysis. For linear advection, Hu et al. [82] have shown that the DG scheme admits one physical mode and  $N$  spurious modes which dissipate quickly for upwind fluxes but remain for central fluxes. Gassner and Kopriva [65] investigated the influence of Legendre-Gauss and Legendre-Gauss-Lobatto integration rules on the dispersion and dissipation characteristics of nodal DG schemes. Moura et al. [139] observed that the spurious modes are in fact replications of the physical mode and contribute to the overall accuracy of the scheme. In particular, for higher wave numbers, the secondary eigenmodes may strongly influence the behavior of the scheme. Furthermore, based on the related flux reconstruction approach, Asthana and Jameson [9] derived a family of schemes with minimal dissipation and dispersion error for advection problems and Vermeire and Vincent [199] investigated the behavior of fully discrete flux reconstruction schemes which includes the influence of the chosen explicit or implicit Runge-Kutta scheme for time integration.

In the above works, with the exception of [138], eigenanalysis of the numerical scheme is based on the linear advection equation. Furthermore, Manzanero et al. [123] develop a generalized von Neumann technique to study the dispersion and dissipation properties of various DG schemes for nonconstant coefficient advection equations. However, parabolic equations have not been extensively studied in this context.

Regarding wave propagation characteristics, the investigation of dissipation and dispersion properties of DG schemes applied to advection-diffusion problems is more recent than for pure advection. For pure diffusion problems, eigenanalysis for both well-resolved and under-resolved cases has been carried out e.g. by Huynh [86] for flux reconstruction schemes and by Alhawwary and Wang [2] for DG schemes. Furthermore, analytic Fourier eigenanalysis of DG diffusion discretizations for the well-resolved regime of wave numbers has been carried out for the DG scheme in [216, 74].

For advection-diffusion problems, an eigenanalysis by Manzanero et al.[122] for DG schemes considers the influence of a parameter-dependent Riemann solver for advective terms and the BR1 scheme for viscous terms. Their study investigates both the individual contribution of the dissipative mechanisms on the whole range of wave numbers and their combined effect.

In addition, the authors correlate their findings to the results for 3D compressible Navier-Stokes flow. Furthermore, Watkins et al. [210] carry out a von Neumann analysis of nodal DG schemes obtained via flux reconstruction, in order to investigate the stability, dissipation and dispersion properties for advection-diffusion problems. In particular, their work analyzes the influence of different interface flux formulations. More precisely, for a DG scheme of polynomial degree  $N = 2$  on Legendre-Gauss nodes, differences of its wave propagation properties are studied in case of either upwind or central flux for advection as well as either a particular alternate LDG scheme or the BR1 approach for diffusion. Their results show that the corresponding schemes with central flux discretization (central flux for advection and BR1 discretization for diffusion) produce smaller errors for well-resolved solutions whereas one-sided flux discretizations (upwind flux combined with LDG discretization) produce smaller errors in the under-resolved case. It is also worth mentioning that the analysis by Watkins et al. [210] combines all eigenmodes into a wave number dependent error measure for the DG approximate solution instead of considering only the dispersion and dissipation properties of a single eigenmode which is regarded as the physical one.

Following the investigations by Zhang and Shu [216] and Guo et al. [74], a comparative Fourier analysis of the Legendre-Gauss and Legendre-Gauss-Lobatto DG schemes has been carried out by the author in [149] for  $N = 1$  using different viscous flux discretizations which confirmed the higher accuracy of Legendre-Gauss integration nodes in the well-resolved regime, i.e. for small wave numbers or small cell sizes. In [148], this analysis is extended to the full range of wave numbers in order to gain additional insight into the numerical behavior for under-resolved waves. Hereby, we use the combined approach of Watkins et al. to obtain a wave number dependent error bound. Extending the study by Watkins et al. [210], DG schemes on Legendre-Gauss-Lobatto nodes are considered as well and a larger range of interface fluxes for diffusion terms is investigated including the BR2 scheme. While the focus of the analysis in [210] was put on the DG scheme for  $N = 2$ , in this work, the combined error measure is computed for a larger range of polynomial degrees to detect odd-even phenomena.

Regarding the wave number dependent error analysis, significant differences can be noted for the classical DG diffusion discretizations LDG and BR1/BR2. The original LDG scheme [44] represents a parameter-dependent family of diffusion discretizations. However, it commonly denotes the choice of alternating fluxes for the reformulated viscous term. For this choice, there are two alternatives, also considered with respect to their superconvergence properties by Cheng and Shu [40]. For advection-diffusion equations, this means that the diffusion fluxes are chosen either consistent with the convective flux, namely the numerical flux for the unknown quantity is chosen equal to the upwind flux while the numerical flux for the gradient is taken from the opposite direction as in (2.9), or the choice of diffusion fluxes is inconsistent with the convective flux when the directions are reversed as in (2.8). While for pure diffusion problems, there is no preferred direction, the LDG scheme for advection-diffusion problems is often preferred in its consistent variant. Although Cheng and Shu [40] show superconvergence properties for both variants, they state a preference for the consistent variant in their conclusions. Furthermore, a recent investigation on IMEX time integration for LDG schemes by Wang et al. [205] also uses the consistent variant. In this work, it is shown that the two choices of alternating fluxes lead to significant differences in the error vs. wave number characteristics. More precisely, the variant which is inconsistent with the upwind advective flux is more accurate in case of advection-diffusion discretizations for a large

range of wave numbers. Furthermore, the investigations in this work show that the higher accuracy of the BR1 flux for well-resolved solutions observed in [210] is restricted to the DG scheme on Legendre-Gauss nodes and to an even polynomial degree  $N$ . For the DG scheme on Legendre-Gauss nodes with odd  $N$ , both the BR2 flux for larger penalty parameters and the more accurate variant of LDG produce smaller errors for well-resolved problems, while for Legendre-Gauss-Lobatto nodes, the inconsistent LDG variant generally performs better as well.

In the following, we first introduce the DG space discretization for linear advection-diffusion equations in one space dimension building on the notation used in Section 2.1. Afterwards, Fourier analysis is introduced to numerically compute the eigensolutions of the respective variants of the DG scheme and derive the corresponding relative error vs. non-dimensional wave number. Numerical experiments for the advection-diffusion problem are then carried out to verify the error analysis.

#### 2.4.1 The DG-discretized linear advection-diffusion equation

We consider the linear advection-diffusion equation

$$\frac{\partial}{\partial t} U(x, t) + a \frac{\partial}{\partial x} U(x, t) = d \frac{\partial^2}{\partial x^2} U(x, t), \quad (x, t) \in Q = \Omega \times (0, T), \quad \Omega = (x_\alpha, x_\beta), \quad (2.35)$$

with diffusion coefficient  $d > 0$  and advective velocity  $a > 0$ , supplemented by the periodic initial condition  $U(x, 0) = U_0(x)$  in  $L^2(\Omega)$  and periodic boundary conditions.

Analogously to the construction of DG schemes for the linear diffusion equation in Section 2.1, the space discretization of (2.35) is derived from the first order reformulation

$$\frac{\partial}{\partial t} U(x, t) + a \frac{\partial}{\partial x} U(x, t) = d \frac{\partial}{\partial x} Q(x, t), \quad Q(x, t) = \frac{\partial}{\partial x} U(x, t), \quad (2.36)$$

introducing the auxiliary variable  $Q$ . Although the analysis of energy stability with respect to the BR2 penalty parameter in the previous section applies to general non-overlapping one-dimensional grids, for the currently presented eigensolution analysis, the computational grid is again assumed to be uniform with constant cell size  $\Delta x$ . Reusing the notation in Section 2.1, the element-wise DG space discretization to obtain the approximate solution  $(u(t), q(t)) \in V_h^2$  is derived from the variational formulation

$$(u_t, v)_j = a \left( (u, v_x)_j - u_{j+1}^- v_{j+1}^- + u_j^- v_j^+ \right) + d \left( -(q, v_x)_j + q_{j+1}^* v_{j+1}^- - q_j^* v_j^+ \right), \quad \forall v \in V_h, \quad (2.37)$$

$$(q, r)_j = -(u, r_x)_j + u_{j+1}^* r_{j+1}^- - u_j^* r_j^+, \quad \forall r \in V_h, \quad (2.38)$$

where  $q^*$  and  $u^*$  represent suitable numerical fluxes determining the chosen DG diffusion scheme and upwind fluxes are applied to the advective term. Therefore, the focus of the eigensolution analysis is put on the influence of the various contemporary diffusion fluxes introduced in Section 2.1). In a similar setting, the influence of central fluxes for advection has been studied in [151].

For the following eigensolution analysis with respect to different diffusion fluxes, numerical integration of the occurring integrals is carried out either exactly using Legendre-Gauss nodes or less accurately with Legendre-Gauss-Lobatto quadrature. Denoting by  $\xi_\nu$ ,  $\nu = 1, \dots, N+1$ , the set of quadrature nodes on the reference cell  $I = [-1, 1]$  and by  $\omega_\nu$ ,  $\nu = 1, \dots, N+1$ , the corresponding quadrature weights, we hence replace the integrals occurring in (2.37), (2.38) by

$$(u, v)_j = \int_{x_j}^{x_{j+1}} uv \, dx \approx \frac{\Delta x}{2} \sum_{\nu=1}^{N+1} \omega_\nu u(\Lambda_j(\xi_\nu)) v(\Lambda_j(\xi_\nu)) = \langle u, v \rangle_j, \quad \text{for } u, v \in V_h. \quad (2.39)$$

The formulation (2.37), (2.38) is thus replaced by

$$\begin{aligned} \langle u_t, v \rangle_j &= a \left( \langle u, v_x \rangle_j - u_{j+1}^- v_{j+1}^- + u_j^- v_j^+ \right) \\ &\quad + d \left( - \langle q, v_x \rangle_j + q_{j+1}^* v_{j+1}^- - q_j^* v_j^+ \right), \end{aligned} \quad \forall v \in V_h, \quad (2.40)$$

$$\langle q, r \rangle_j = - \langle u, r_x \rangle_j + u_{j+1}^* r_{j+1}^- - u_j^* r_j^+, \quad \forall r \in V_h. \quad (2.41)$$

Analogously to the derivation of the matrix-vector formulation of the DG scheme in Section 1.2.1, applying partial integration in space to the terms  $\langle u, v_x \rangle_j$ ,  $\langle q, v_x \rangle_j$  in (2.40) and  $\langle u, r_x \rangle_j$  in (2.41), we obtain the cell-wise strong form of the DG semi-discretization on the reference cell as follows. Hereby, we also recall that the above terms are exactly integrated in case of Legendre-Gauss-Lobatto quadrature. We collect the nodal values of the approximate solution at the  $N+1$  quadrature points  $\Lambda_j(\xi_\nu)$  within a DG cell (using either Legendre-Gauss or Legendre-Gauss-Lobatto nodes) into the solution vector  $\mathbf{u} = (u_1, \dots, u_{N+1})^T$ , i.e.  $u_\nu \approx U(\Lambda_j(\xi_\nu), t)$ . Furthermore, the Lagrange polynomials  $L_k(\xi)$  corresponding to the nodal set are used as the DG test and basis functions and collected into the vector valued function  $\mathbf{L}$  given by  $\mathbf{L}(\xi) = (L_1(\xi), \dots, L_{N+1}(\xi))^T$ . Defining the differentiation matrix  $\mathbf{D}$  and discrete mass matrix  $\mathbf{M}$  by their entries  $D_{jk} = L'_k(\xi_j)$  and  $M_{jk} = \delta_{jk}\omega_j = M_{kj}$ , the resulting DG formulation reads

$$\frac{\Delta x}{2} \mathbf{u}_t + a \mathbf{D} \mathbf{u} - d \mathbf{D} \mathbf{q} = \mathbf{M}^{-1} (a[(u - u^-) \mathbf{L}]_{-1}^1 - d[(q - q^*) \mathbf{L}]_{-1}^1), \quad (2.42)$$

$$\frac{\Delta x}{2} \mathbf{q} - \mathbf{D} \mathbf{u} = -\mathbf{M}^{-1} [(u - u^*) \mathbf{L}]_{-1}^1, \quad (2.43)$$

where  $q^*$  and  $u^*$  again represent the numerical diffusion fluxes.

In the above formulation, we also recall that in case of Legendre-Gauss-Lobatto nodes, we have a lumped mass matrix  $\mathbf{M}$  with  $M_{jk} \approx \int_{-1}^1 L_j L_k d\xi$  while for Legendre-Gauss nodes, integration is exact, i.e.  $M_{jk} = \int_{-1}^1 L_j L_k d\xi$ .

## 2.4.2 Eigensolution analysis

In the following, the accuracy of the DG schemes employing different diffusion fluxes is studied via Fourier analysis. The linear advection-diffusion equation (2.35) admits a traveling wave solution of the form  $u(x, t) = e^{i(kx - \omega t)}$  where  $k \in \mathbb{R}$  is the wave number and  $\omega$  denotes the frequency given by  $\omega = ak - idk^2$  for the analytical solution.

Inserting a corresponding numerical solution of the form  $\mathbf{u}_j(t) = \mathbf{c} e^{i(kj\Delta x - \tilde{\omega}t)}$  into the strong DG formulation (2.42), (2.43) yields

$$\begin{aligned} & \left( a \left( \mathbf{A}_0 + e^{-ik\Delta x} \mathbf{A}_{-1} \right) + \frac{d}{\Delta x} \left( e^{-2ik\Delta x} \mathbf{B}_{-2} + e^{-ik\Delta x} \mathbf{B}_{-1} + \mathbf{B}_0 + e^{ik\Delta x} \mathbf{B}_1 + e^{2ik\Delta x} \mathbf{B}_2 \right) \right) \mathbf{c} \\ &= i\Delta x \tilde{\omega} \mathbf{c}, \end{aligned} \quad (2.44)$$

with real  $(N+1) \times (N+1)$  matrices  $\mathbf{A}_0, \mathbf{A}_{-1}, \mathbf{B}_k, k = 0, \pm 1, \pm 2$ , depending only on the chosen nodal set and the diffusion fluxes  $u^*, q^*$  which characterize the respective DG scheme. Numerical solutions of the prescribed form can thus be found by solving the eigenvalue problem (2.44). We may furthermore reduce the set of parameters by defining the non-dimensional wave number  $K = k\Delta x$ , the non-dimensional numerical frequency  $\Omega = \frac{\Delta x \tilde{\omega}}{a}$  and the grid Peclet number  $Pe^* = \frac{a\Delta x}{d}$  to obtain the eigenvalue problem

$$\left( (\mathbf{A}_0 + e^{-iK} \mathbf{A}_{-1}) + \frac{1}{Pe^*} (e^{-2iK} \mathbf{B}_{-2} + e^{-iK} \mathbf{B}_{-1} + \mathbf{B}_0 + e^{iK} \mathbf{B}_1 + e^{2iK} \mathbf{B}_2) \right) \mathbf{c} = i\Omega \mathbf{c}. \quad (2.45)$$

For a polynomial degree of  $N$ , for any  $K \in [0, \pi(N+1)]$ , we obtain a set of  $N+1$  eigenvalues of equation (2.45). In the literature, see e.g. Hu et al. [82], often only one of these is considered physical and the function assigning to each  $K$  its corresponding physical eigenvalue is termed the physical mode while the remaining modes are named spurious.

### Dispersion and dissipation properties depending on the diffusion fluxes

Figures 2.1a and 2.1c plot the real and imaginary part of the three modes for the DG( $N = 2$ ) scheme on Legendre-Gauss nodes with LDG diffusion flux for a grid Peclet number of  $Pe^* = 20$ . For both variants of LDG, the eigenvalues coincide. Figures 2.1b and 2.1d show the eigenvalue curves for the LDG diffusion flux employed within the DG scheme on Legendre-Gauss-Lobatto nodes while Figure 2.2 depicts the results for the BR1 and BR2 diffusion fluxes on Legendre-Gauss and Legendre-Gauss-Lobatto nodes for  $N = 2$ , respectively. For the BR2 scheme, the penalty parameter is hereby set to  $\eta_e = 3$  and on Legendre-Gauss-Lobatto nodes the BR2<sub>LGL</sub> variant is implemented, whereby the BR2 lifting operator is computed using (2.12). Comparing the choice of DG nodal set in Figures 2.1 and 2.2, we see that for the more accurate Legendre-Gauss quadrature rule, the physical mode stays close to the exact dispersion relation for a larger range of wave numbers in all cases. For the BR1 scheme and the BR2 scheme with  $\eta_e = 3$ , the numerical dispersion relations depicted in Figure 2.2 differ. In particular, the BR2 approach introduces more numerical dissipation to the physical mode for higher wave numbers as well as to the spurious modes due to the comparatively high value of  $\eta_e$ . Figure 2.3 shows the corresponding results when varying the penalty parameter  $\eta_e$  of the BR2 fluxes in case of the DG scheme on Legendre-Gauss nodes. Reducing  $\eta_e$  yields results closer to BR1. In fact, for  $\eta_e = \frac{N}{N+1} = \frac{2}{3}$ , i.e. the smallest value of the penalty parameter yielding a provable energy stable scheme as described in Section 2.3, the eigenvalue curves are almost indistinguishable from those of the BR1 scheme. This behavior is not surprising, since the smallest choice of the BR2 penalty parameter in case of the DG scheme on Legendre-Gauss-Lobatto nodes is equivalent to BR1, see Theorem 2.7.

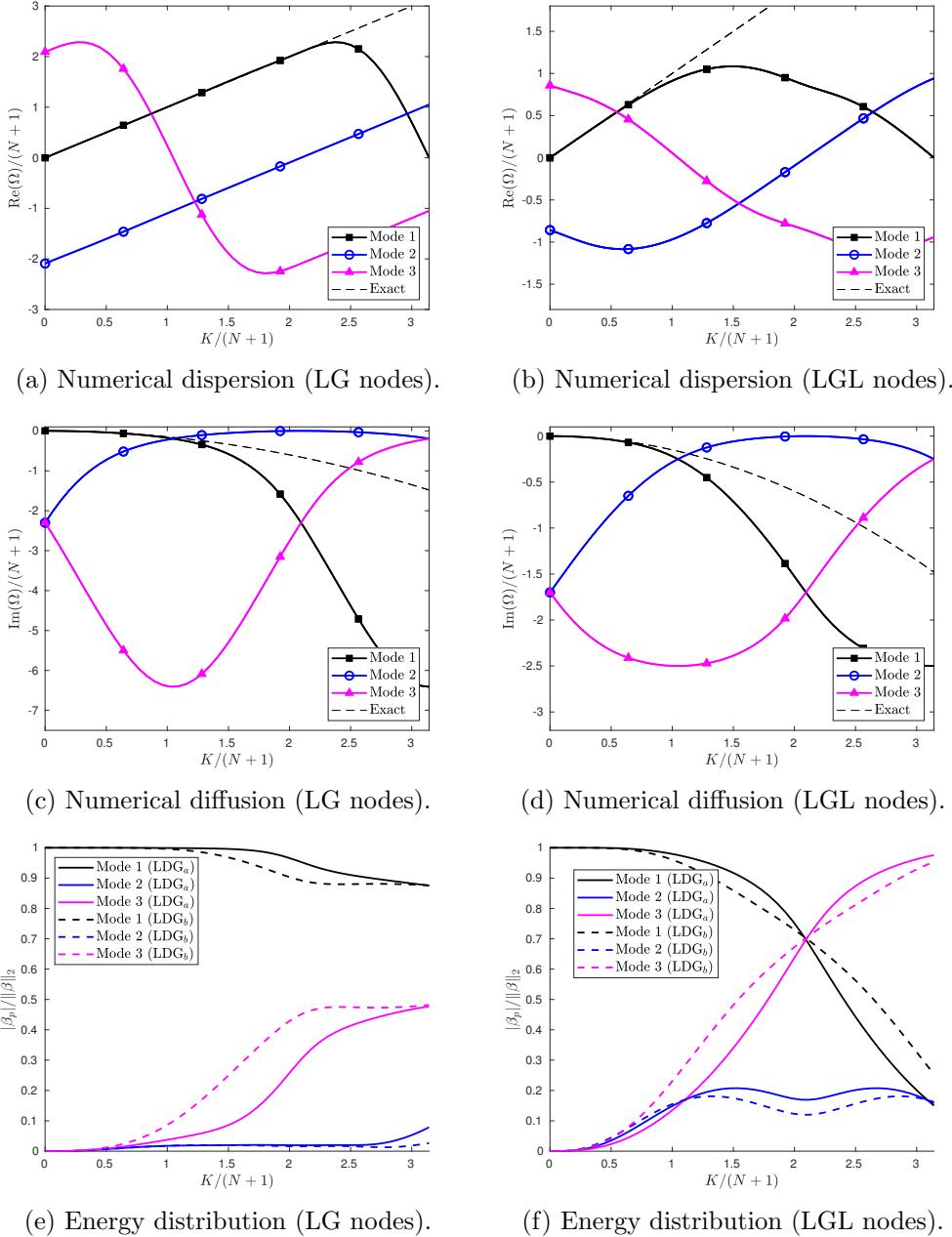


Figure 2.1: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 2$  using Legendre-Gauss (LG) nodes (first column) and Legendre-Gauss-Lobatto (LGL) nodes (second column). Diffusion term discretized by the two alternate variants of the LDG diffusion, LDG<sub>a</sub> (solid lines) and LDG<sub>b</sub> (dashed lines).

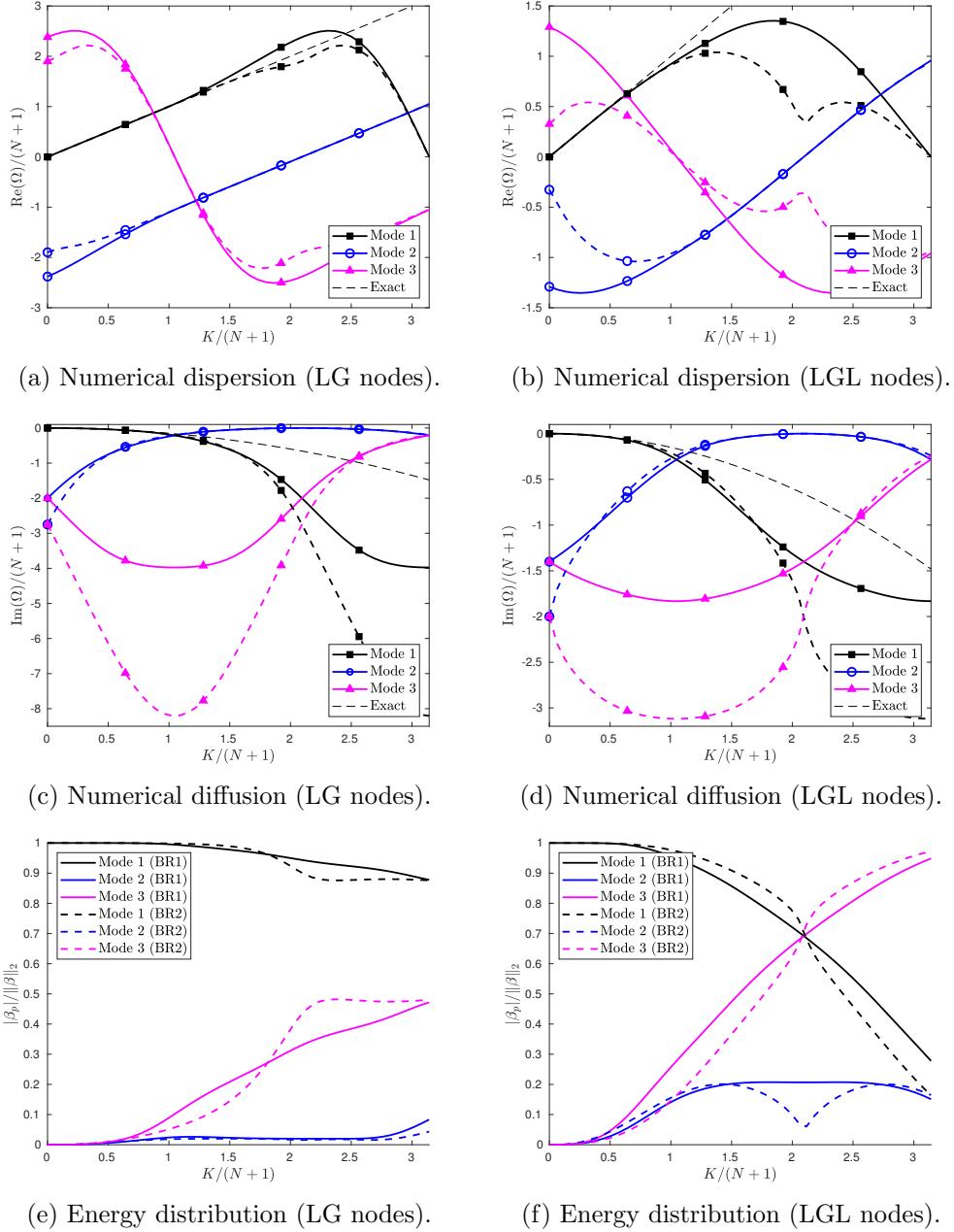


Figure 2.2: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 2$  using Legendre-Gauss (LG) nodes (first column) and Legendre-Gauss-Lobatto (LGL) nodes (second column). Diffusion term discretized by BR1 scheme (solid lines) and BR2 method with  $\eta_e = 3$  (dashed lines).

### Energy distribution of the eigenmodes

Solely regarding the physical mode does not yield a complete picture of the behavior of the numerical scheme. In fact, Moura et al. [139] argue that the spurious modes contribute to the overall accuracy, in particular for higher wave numbers where the secondary eigenmodes may strongly influence the behavior of the scheme. Also deviating from the distinction of a physical mode, Watkins et al. [210] combine all eigensolutions into an error bound depending on the wave number. We will follow their work which also allows us to point out differences between the two variants of the LDG approach. For this purpose, we regard the complete set of  $N + 1$  eigenvalues  $\Omega_p$ ,  $p = 1, \dots, N + 1$ , and corresponding normalized eigenvectors  $\mathbf{v}_p$ ,  $p = 1, \dots, N + 1$ .

An initial wave with non-dimensional wave number  $K$  on a cell  $I_j$ , given by the initial nodal values

$$u_\nu(0) = e^{i(j+\frac{\xi\nu+1}{2})K}, \quad \nu = 1, \dots, N + 1, \quad (2.46)$$

can be represented as the linear combination  $\mathbf{u}(0) = \sum_{p=1}^{N+1} \beta_p \mathbf{v}_p e^{ijK}$ , where the coefficients  $\beta_p$  are obtained as the solution of  $\sum_{p=1}^{N+1} \beta_p \mathbf{v}_p = \boldsymbol{\alpha}$ , with  $\alpha_\nu = e^{i\frac{\xi\nu+1}{2}K}$ . Then, the corresponding numerical solution of the DG scheme (2.42), (2.43) on the cell  $I_j$  is a linear combination of the waves  $\mathbf{v}_p e^{i(jK - \frac{a}{\Delta x} \Omega_p t)}$ ,  $p = 1, \dots, N + 1$ . All eigenmodes are therefore present in the numerical solution of the DG scheme for the above initial condition. As also stated by Watkins et al. [210], the weights  $\beta_p$  determine how each mode contributes quantitatively to the numerical solution and the normalized weights  $\frac{|\beta_p|}{\|\boldsymbol{\beta}\|_2}$  describe the distribution of energy among the modes as depicted in Figures 2.1e, 2.1f, 2.2e and 2.2f.

Hereby, Figures 2.1e and 2.1f show that although both variants of LDG have the same eigenmodes, the energy distribution among those modes is different for  $\text{LDG}_a$  and  $\text{LDG}_b$ . More precisely, for moderate wave numbers the first eigenmode – which is often considered the physical one – has more influence on the numerical solution for  $\text{LDG}_a$  than for  $\text{LDG}_b$ . In addition, in this wave number regime, the spurious mode 3 has less influence for  $\text{LDG}_a$ . Furthermore, Figure 2.2 depicts differences of both the eigenmodes and their energy distribution comparing the BR1 and BR2 diffusion discretization. Here, for low to moderate wave numbers, the BR2 scheme with  $\eta_e = 3$  leads to a higher energy content of the physical mode while for high wave numbers, the energy content of the physical mode is higher for the BR1 scheme. In addition, the BR2 approach introduces a larger amount of numerical diffusion for the spurious mode 3 for this value of  $\eta_e$  as shown in Figures 2.2c and 2.2d. This mode still has a significant energy content as shown in Figures 2.2e and 2.2f and has therefore a considerable impact on the behavior of the numerical solution. The dependency of the numerical dissipation and the energy content of each mode on the BR2 penalty parameter  $\eta_e$  is indicated in Figure 2.3.

Increasing the polynomial degree, Figures 2.4 and 2.5 show the four modes for the  $\text{DG}(N = 3)$  scheme. Considering the two variants of alternating LDG fluxes, for moderate wave numbers the first eigenmode has again more influence on the numerical solution for  $\text{LDG}_a$  than for  $\text{LDG}_b$ , while the influence of the spurious mode 4 is significantly reduced for  $\text{LDG}_a$ . Considering the comparison of the BR1 scheme to the BR2 scheme with parameter  $\eta_e = 3$ , differences regarding the DG nodal set can be perceived. On Legendre-Gauss nodes, the results are similar to the  $\text{DG}(N = 2)$  scheme relating to the higher energy content of the physical mode

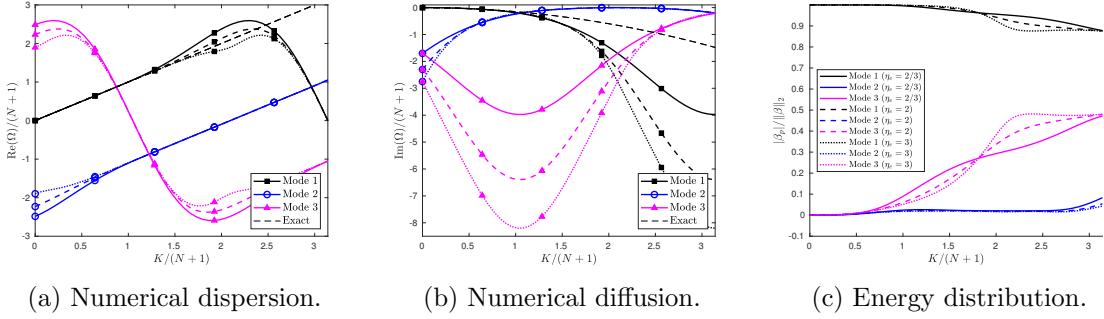


Figure 2.3: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 2$  using Legendre-Gauss nodes. Diffusion term discretized by the BR2 scheme for  $\eta_e = 2/3$  (solid lines),  $\eta_e = 2$  (dashed lines) and  $\eta_e = 3$  (dotted lines).

for the BR2 scheme in case of moderate wave numbers and of the BR1 scheme for high wave numbers. However, on Legendre-Gauss-Lobatto nodes, the BR2 scheme yields higher energy content of the physical mode for high wave numbers as well.

### A wave number dependent error bound combining all eigenmodes

As in Watkins et al. [210], we now consider the error of the numerical solution depending on the non-dimensional wave number. Hereby, we again consider the above initial condition (2.46). The numerical solution on cell  $I_j$  corresponding to this initial condition is

$$\mathbf{u}(t) = \sum_{p=1}^{N+1} \beta_p \mathbf{v}_p e^{i(jK - \frac{a}{\Delta x} \Omega_p t)}, \quad (2.47)$$

whereas the exact solution is given by

$$\mathbf{u}^{ex}(t) = \sum_{p=1}^{N+1} \beta_p \mathbf{v}_p e^{ijK - \frac{a}{\Delta x} (iK + (Pe^*)^{-1} K^2) t}. \quad (2.48)$$

For the absolute error of the numerical solution, we therefore have

$$\mathbf{e}(t) = \mathbf{u}(t) - \mathbf{u}^{ex}(t) = e^{ijK} e^{-\frac{a}{\Delta x} (iK + (Pe^*)^{-1} K^2) t} \sum_{p=1}^{N+1} \left( e^{\frac{a}{\Delta x} (-i(\Omega_p - K) + (Pe^*)^{-1} K^2) t} - 1 \right) \beta_p \mathbf{v}_p. \quad (2.49)$$

Using

$$\begin{aligned} \|\mathbf{u}^{ex}(t)\|_{l2} &= e^{-\frac{a}{\Delta x} (Pe^*)^{-1} K^2 t} \|\mathbf{u}(0)\|_{l2} = e^{-\frac{a}{\Delta x} (Pe^*)^{-1} K^2 t} \sqrt{\sum_{\nu=1}^{N+1} \left| e^{i \frac{\xi_\nu + 1}{2} K} \right|^2} \\ &= e^{-\frac{a}{\Delta x} (Pe^*)^{-1} K^2 t} \sqrt{N+1}, \end{aligned}$$

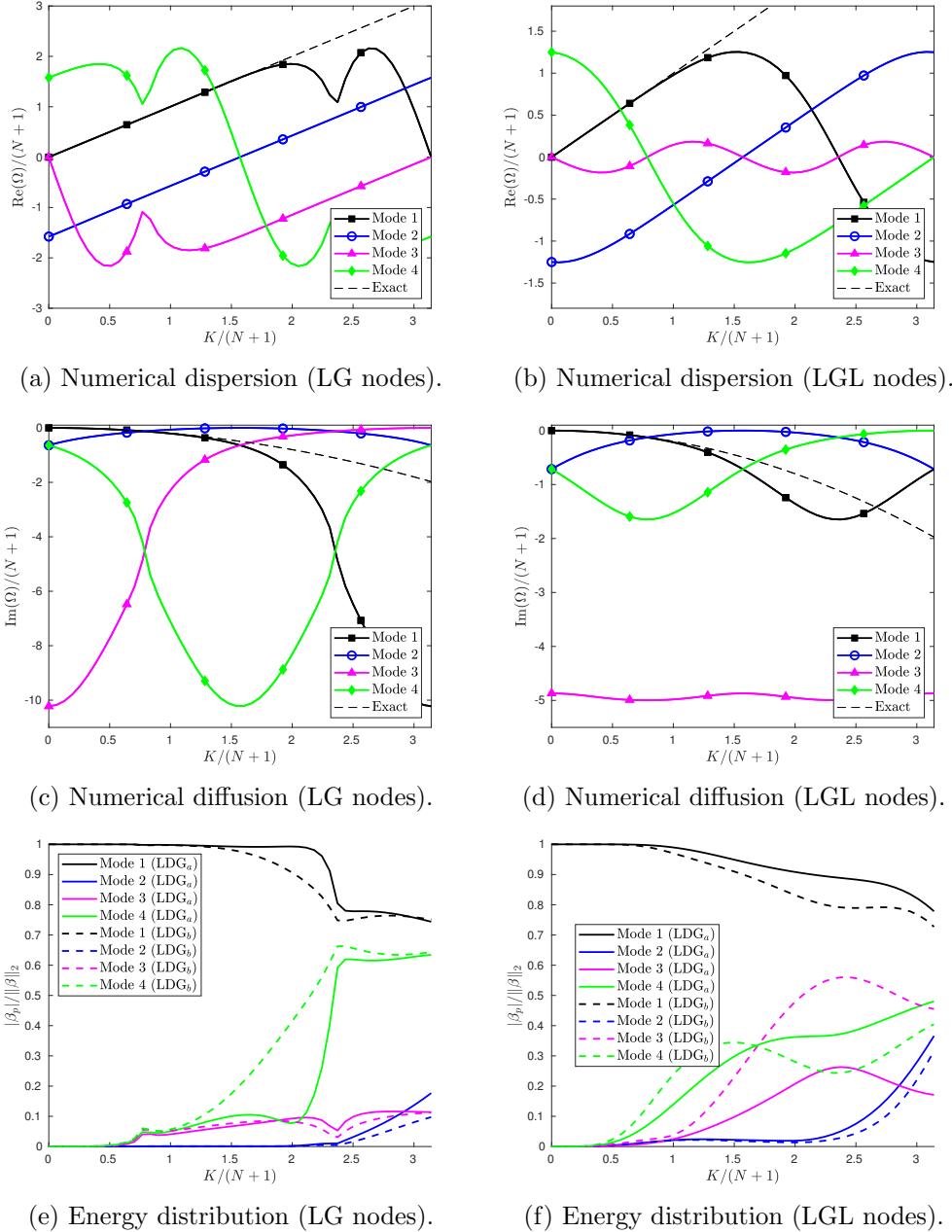


Figure 2.4: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 3$  using Legendre-Gauss (LG) nodes (first column) and Legendre-Gauss-Lobatto (LGL) nodes (second column). Diffusion term discretized by the two alternate variants of the LDG diffusion, LDG<sub>a</sub> (solid lines) and LDG<sub>b</sub> (dashed lines).

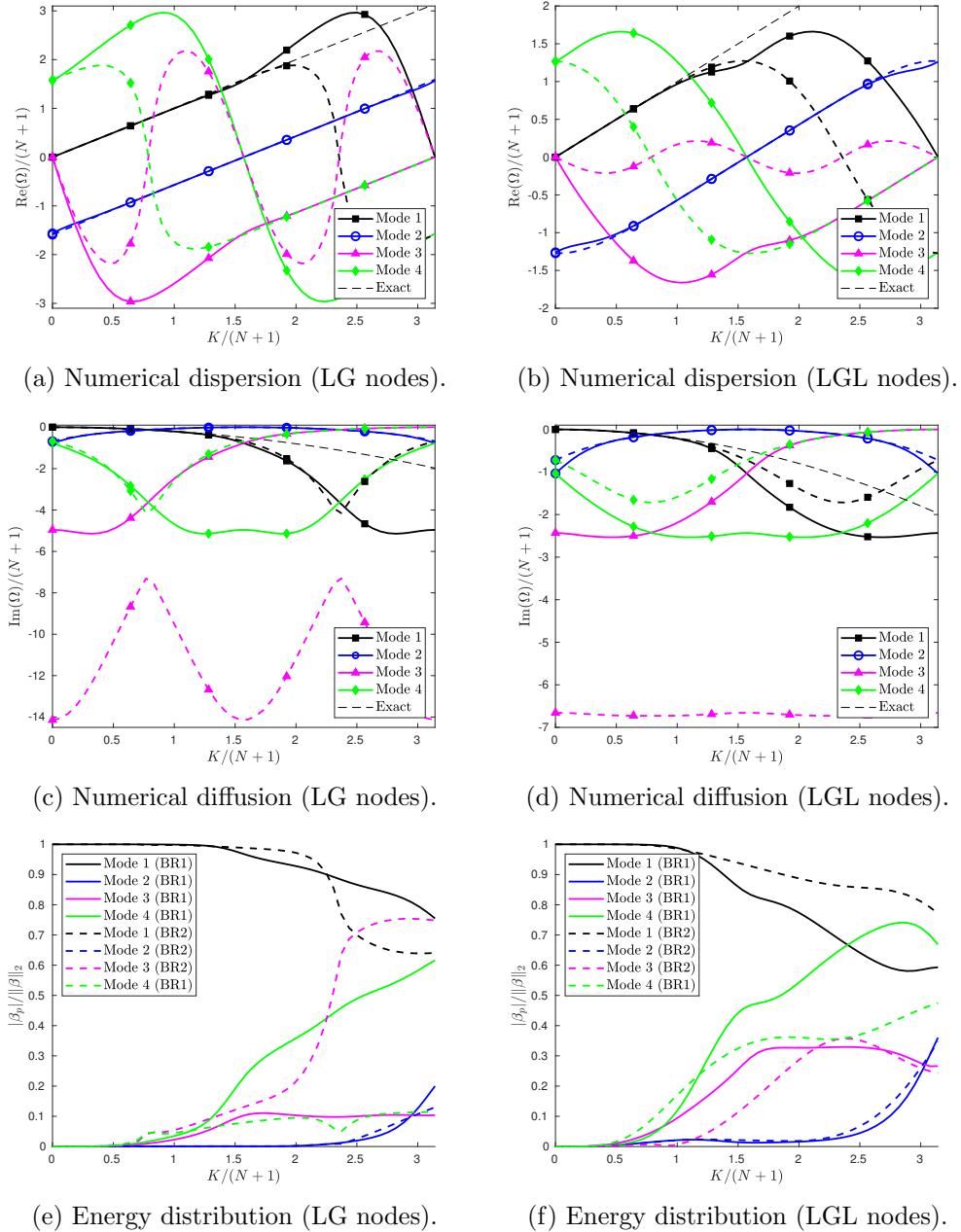


Figure 2.5: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 3$  using Legendre-Gauss (LG) nodes (first column) and Legendre-Gauss-Lobatto (LGL) nodes (second column). Diffusion term discretized by BR1 scheme (solid lines) and BR2 method with  $\eta_e = 3$  (dashed lines).

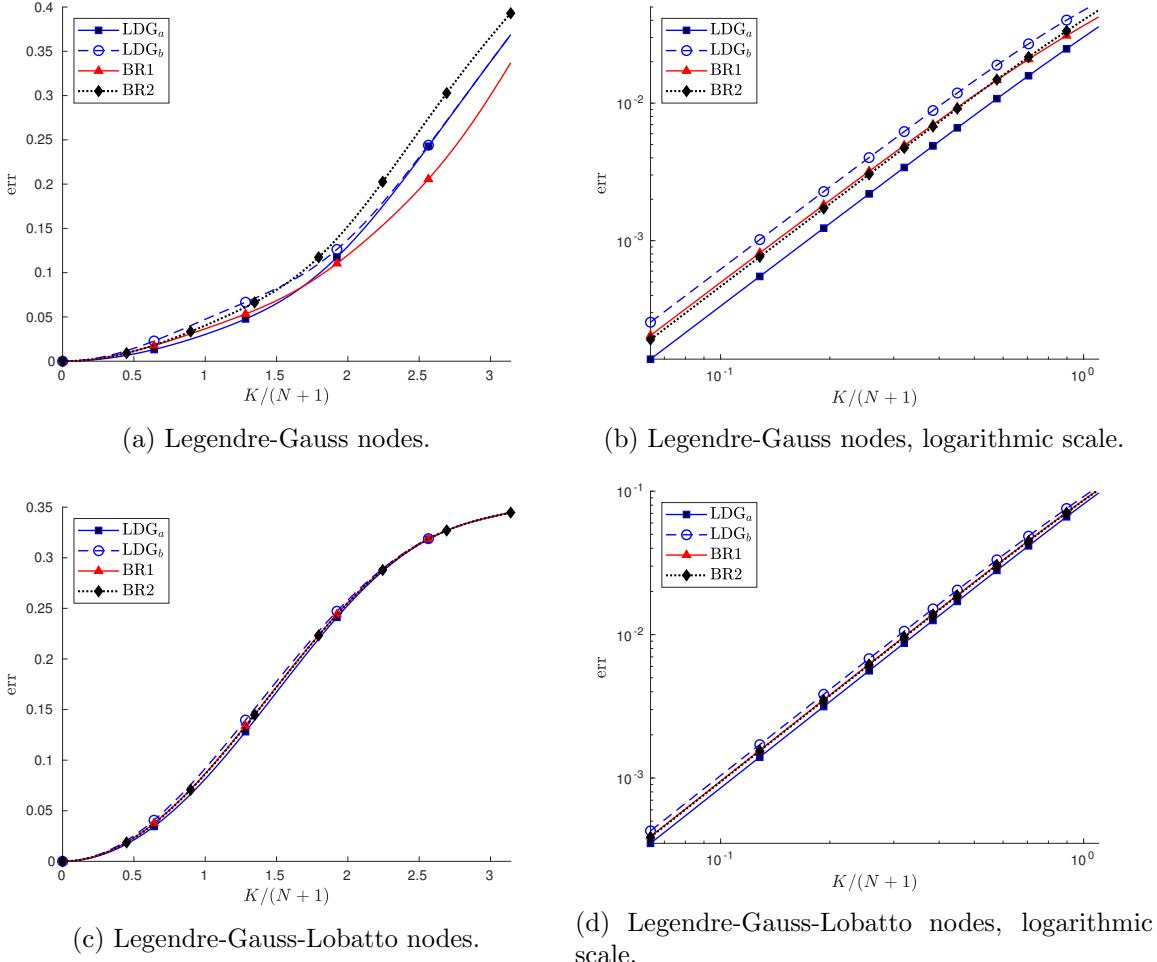


Figure 2.6: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for  $N = 1$ ,  $Pe^* = 20$ ,  $t = 0.05$ .

the norm of the relative error of the numerical solution is therefore given by

$$\text{err}(t) := \frac{\|\mathbf{e}(t)\|_{l^2}}{\|\mathbf{u}^{ex}(t)\|_{l^2}} = \frac{1}{\sqrt{N+1}} \left\| \sum_{p=1}^{N+1} \left( e^{\frac{a}{\Delta x}(-i(\Omega_p - K) + (Pe^*)^{-1}K^2)t} - 1 \right) \beta_p \mathbf{v}_p \right\|_{l^2}. \quad (2.50)$$

While Watkins et al. [210] use an upper bound of the above quantity (2.50), we will use the exact relative error with respect to the wave number in the following analysis. Particularly for well-resolved wave numbers, a combined error measure is advantageous since in the depiction of eigenmodes the variations with respect to different schemes are almost indistinguishable. In addition, the behavior of different schemes concerning each separate eigenmode is not as perceptible for higher polynomial degrees.

Figures 2.6, 2.7, 2.8 and 2.9 show the relative error as defined in (2.50) at a given time  $t$  versus the non-dimensional wave number for the DG schemes up to 5th order using different

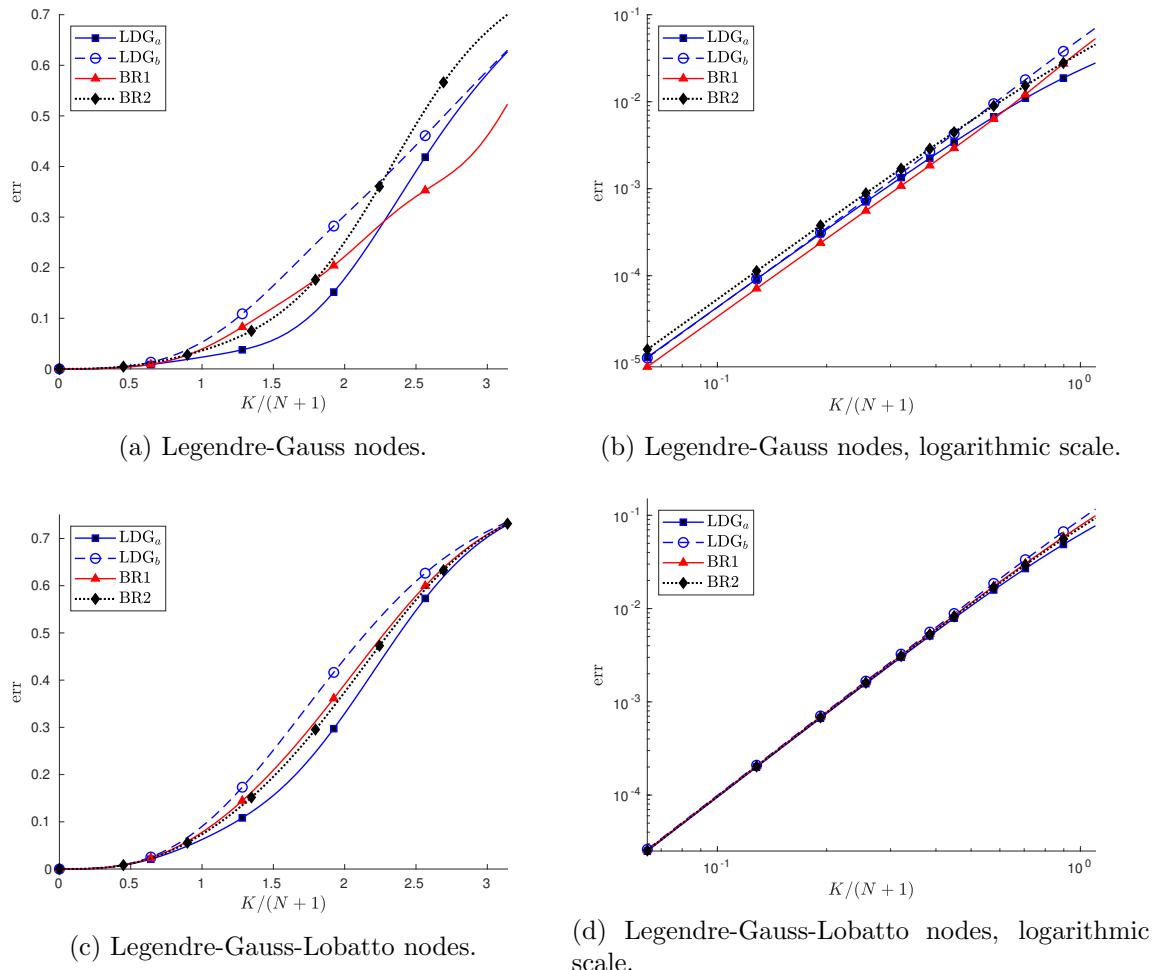


Figure 2.7: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for  $N = 2$ ,  $Pe^* = 20$ ,  $t = 0.05$ .

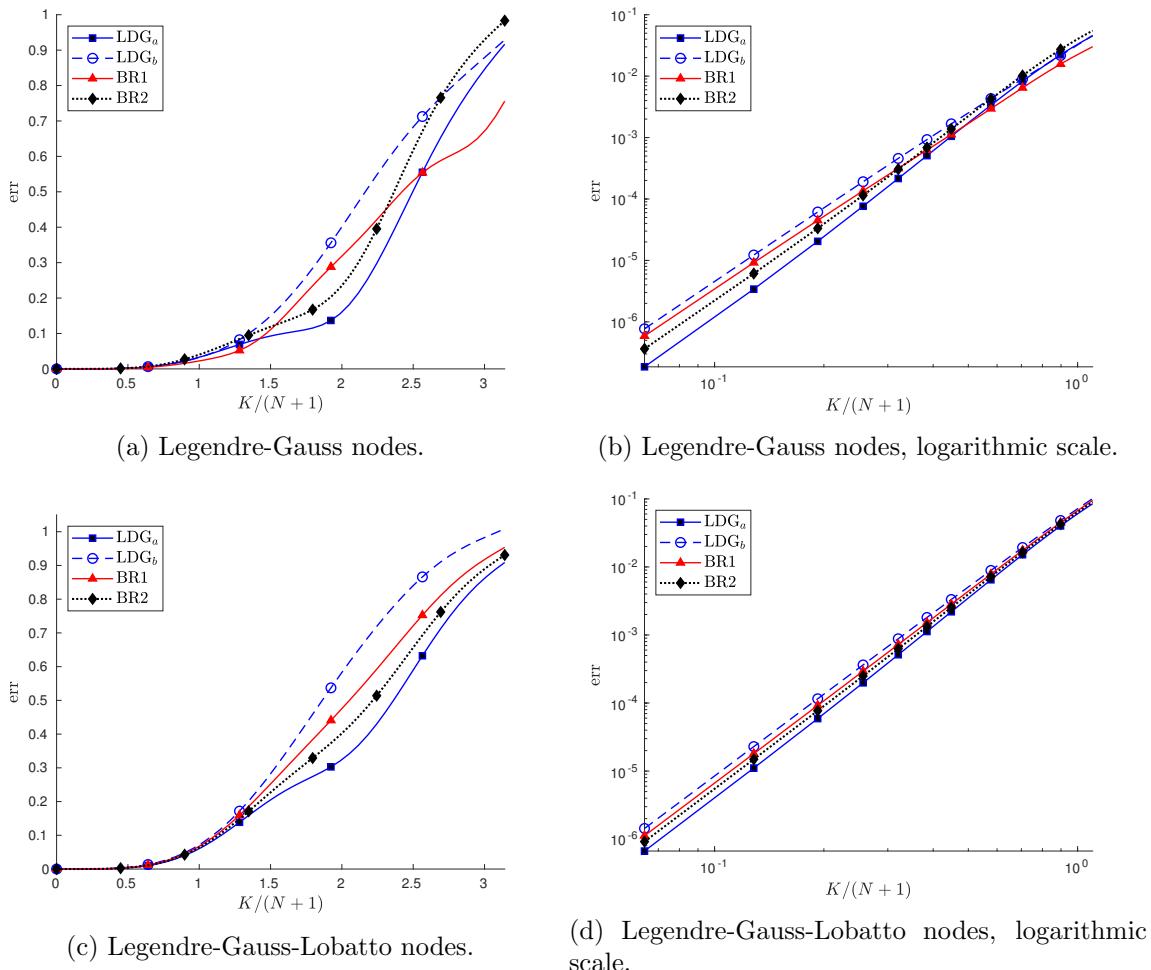


Figure 2.8: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for  $N = 3$ ,  $Pe^* = 20$ ,  $t = 0.05$ .

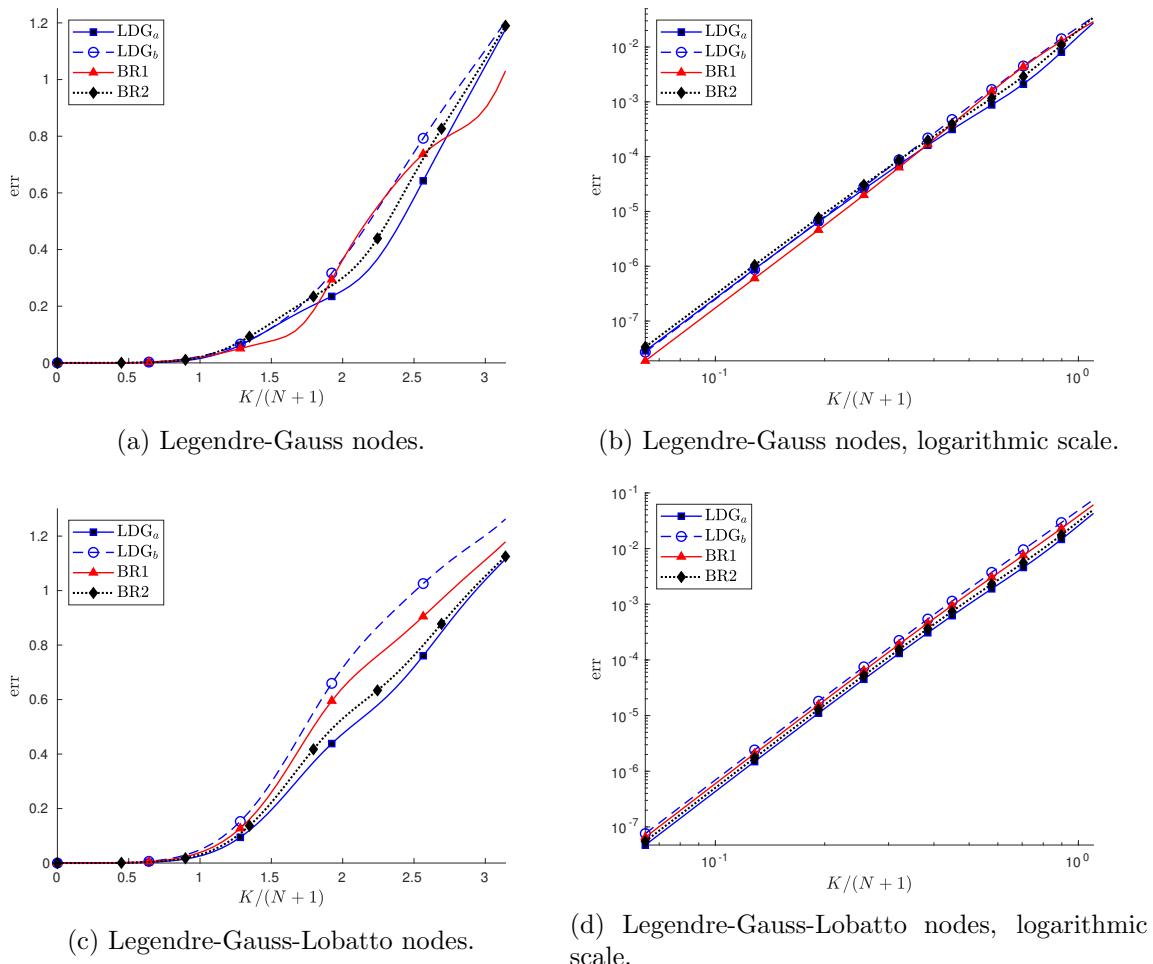


Figure 2.9: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for  $N = 4$ ,  $Pe^* = 20$ ,  $t = 0.05$ .

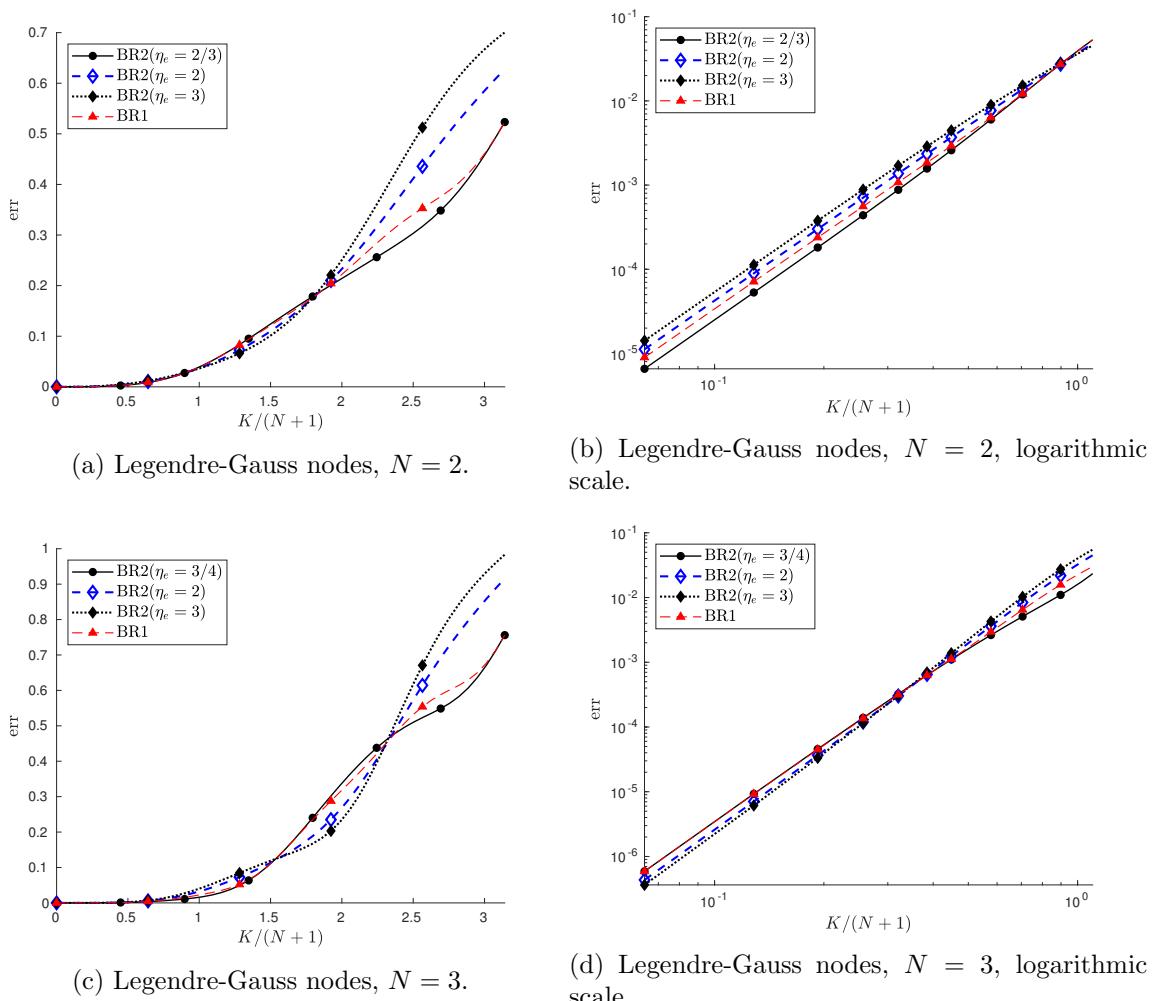


Figure 2.10: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for the BR2 scheme depending on the penalty parameter  $\eta_e$  in comparison to BR1. DG scheme on Legendre-Gauss nodes for  $N = 2$  (first row) and  $N = 3$  (second row).

diffusion fluxes. In particular, the behavior of the alternate LDG<sub>a</sub> and LDG<sub>b</sub> fluxes, the BR1 scheme and the BR2 scheme for  $\eta_e = 3$  are compared. In addition, in Figure 2.10, a study with respect to the BR2 penalty parameter is carried out for the DG( $N = 2$ ) and DG( $N = 3$ ) schemes on Legendre-Gauss nodes.

Considering the implemented variants of LDG, the two choices of alternating fluxes lead to significant differences with respect to the relative error which is due to the fact that the energy distribution among modes differs. More precisely, the variant which is inconsistent with the upwind advective flux is more accurate in case of advection-diffusion discretizations for a large range of wave numbers.

Furthermore, while higher accuracy of the BR1 flux for well-resolved solutions has been observed in [210], the plots in logarithmic scale at the right-hand side of each figure indicate that for very low wave numbers, this higher accuracy is restricted to the DG scheme on Legendre-Gauss nodes with even polynomial degree  $N$ . For the DG scheme on Legendre-Gauss-Lobatto nodes as well as for odd  $N$ , both the BR2 flux with penalty parameter  $\eta_e = 3$  and the more accurate variant of LDG produce a smaller error for well-resolved solutions.

Also for the DG scheme on Legendre-Gauss-Lobatto nodes, Figures 2.7, 2.8 and 2.9 for polynomial degrees of  $N = 2, 3, 4$ , respectively, show the same behavior of the investigated schemes with respect to accuracy. For almost the whole range of wave numbers, the LDG<sub>a</sub> scheme is the most accurate approach, followed by the BR2 scheme with penalty parameter  $\eta_e = 3$ , the BR1 scheme and lastly the LDG<sub>b</sub> scheme. For  $N = 1$  the error curves in Figure 2.6c are nearly indistinguishable but suggest a similar sequence for the schemes.

Concerning the BR2 penalty parameter  $\eta_e$ , Figure 2.10 depicts the relative errors of the DG approximate solutions on Legendre-Gauss nodes for polynomial degrees of  $N = 2, 3$  in case of the parameter choices  $\eta_e = \frac{N}{N+1}, 2, 3$ . For comparison, the results using the BR1 diffusion fluxes are included in the error plots. Obviously, the relative error of the BR2 scheme is closest to the relative error of the BR1 scheme for  $\eta_e = \frac{N}{N+1}$ .

At this point, it should be remarked that even if the results for BR1 and BR2( $\eta_e = \frac{N}{N+1}$ ) are similar, they do not coincide. In fact, on Legendre-Gauss nodes, the BR1 scheme is not equivalent to BR2 for any value of the penalty parameter  $\eta_e$ , since on this set of nodes, the BR1 diffusion discretization has a wider stencil than the BR2 and IP schemes. This does not contradict Theorem 2.7, since its assertions only apply to the DG scheme on Legendre-Gauss-Lobatto nodes.

Furthermore, with respect to the parameter study for  $\eta_e$ , it can be observed that the schemes BR2( $\eta_e = \frac{N}{N+1}$ ) and BR2( $\eta_e = 3$ ) provide lower and upper bounds for the BR1 results as well as for BR2( $\eta_e = 2$ ). Along the whole range from low to high wave numbers, the BR2 schemes for  $\eta_e = \frac{N}{N+1}$  and  $\eta_e = 3$  alternate to provide the most accurate solution among the schemes considered in Figure 2.10. Therefore, there is no optimal choice for  $\eta_e$  regarding accuracy for the whole range of low to high wave numbers. However, we observe that for very high wave numbers, the BR2( $\eta_e = \frac{N}{N+1}$ ) and BR1 schemes yield the most accurate results for both values of  $N$  while for well-resolved wave numbers, the most accurate scheme depends on the polynomial degree. For even polynomial degree  $N = 2$ , the parameter  $\eta_e = \frac{N}{N+1}$  yields the most accurate and  $\eta_e = 3$  the less accurate result whereas for odd polynomial degree  $N = 3$  the roles of the particular BR2 schemes are reversed and BR2( $\eta_e = 3$ ) is provides the best

result. This odd-even phenomenon is investigated more thoroughly by carrying out numerical experiments for the full advection-diffusion problem in Section 2.4.3.

### An example regarding the influence of the grid Peclet number

In order to give an example of the influence of the grid Peclet number on the results of the eigensolution analysis, in particular for higher order DG schemes, Figure 2.11 shows the eigenmodes for the two alternate variants of the LDG diffusion fluxes within the DG( $N = 5$ ) scheme on Legendre-Gauss nodes for  $Pe^* = 20$  as well as  $Pe^* = 100$ . In addition, Figure 2.12 gives a comparison of the numerical errors depending on the non-dimensional wave number. Although the structure of the eigensolutions for the LDG fluxes differs for different Peclet numbers, with a high frequency mode for  $Pe^* = 20$  and replications of the physical mode for  $Pe^* = 100$ , the basic observations regarding the differences with respect to diffusion discretizations still hold. Regarding the LDG variants, for moderate wave numbers, the first eigenmode has more influence on the numerical solution for  $\text{LDG}_a$  than for  $\text{LDG}_b$ . In addition, Figure 2.12 shows that  $\text{LDG}_a$  is more accurate than  $\text{LDG}_b$ . Regarding the results with respect to the BR1 scheme and the BR2 scheme for  $\eta_e = 3$ , which are also depicted in Figure 2.12, the error plots in logarithmic scale indicate a smaller error for the BR2 scheme with penalty parameter  $\eta_e = 3$  in comparison to the BR1 diffusion fluxes for both Peclet numbers in accordance to the observed reduction of accuracy for the BR1 scheme for odd polynomial degrees. A comparison of Figures 2.12a and 2.12c shows that for both Peclet numbers, the BR1 scheme is alternately the most or least accurate scheme depending on the wave number. A different behavior depending on the grid Peclet number can be perceived for the  $\text{LDG}_b$  scheme which produces comparatively larger errors in the higher wave number range for  $Pe^* = 100$ . Naturally, the differences between the diffusion schemes are generally smaller for higher Peclet numbers since then the influence of advection dominates regarding the underlying PDE. In that respect, the  $\text{LDG}_a$  scheme and the BR2 scheme for  $\eta_e = 3$  produce nearly identical results in Figures 2.12c and 2.12d for  $Pe^* = 100$ .

### 2.4.3 Numerical results for the linear advection-diffusion equation

In this section, numerical experiments are carried out to solve the advection-diffusion problem (2.35) for  $a = 1$  on the spatial domain  $\Omega = [-10, 10]$  discretized by  $E = 20$  elements, i.e the grid size is  $\Delta x = 1$ . The diffusion coefficient  $d$  varies depending on the respective test case and can be obtained from the grid Peclet number, which in this case is  $Pe^* = \frac{a\Delta x}{d} = d^{-1}$ . We consider periodic initial conditions and supplement (2.35) with periodic boundary conditions.

Now, the DG scheme (2.42), (2.43) is applied to this periodic advection-diffusion problem for polynomial degrees of  $N = 1, \dots, 5$ . For time integration, the classical explicit fourth order Runge-Kutta scheme is used. Given the DG solution  $\mathbf{u}(t)$  and the exact solution  $\mathbf{u}^{ex}(t)$ , we measure the relative  $L^2$ -error by

$$\text{err}_{L^2}(t) = \|\mathbf{u}(t) - \mathbf{u}^{ex}(t)\|_{L^2} / \|\mathbf{u}^{ex}(t)\|_{L^2}. \quad (2.51)$$

Hereby, the  $L^2$ -norm of a nodal quantity  $\mathbf{v}$  is approximated using the given quadrature rules

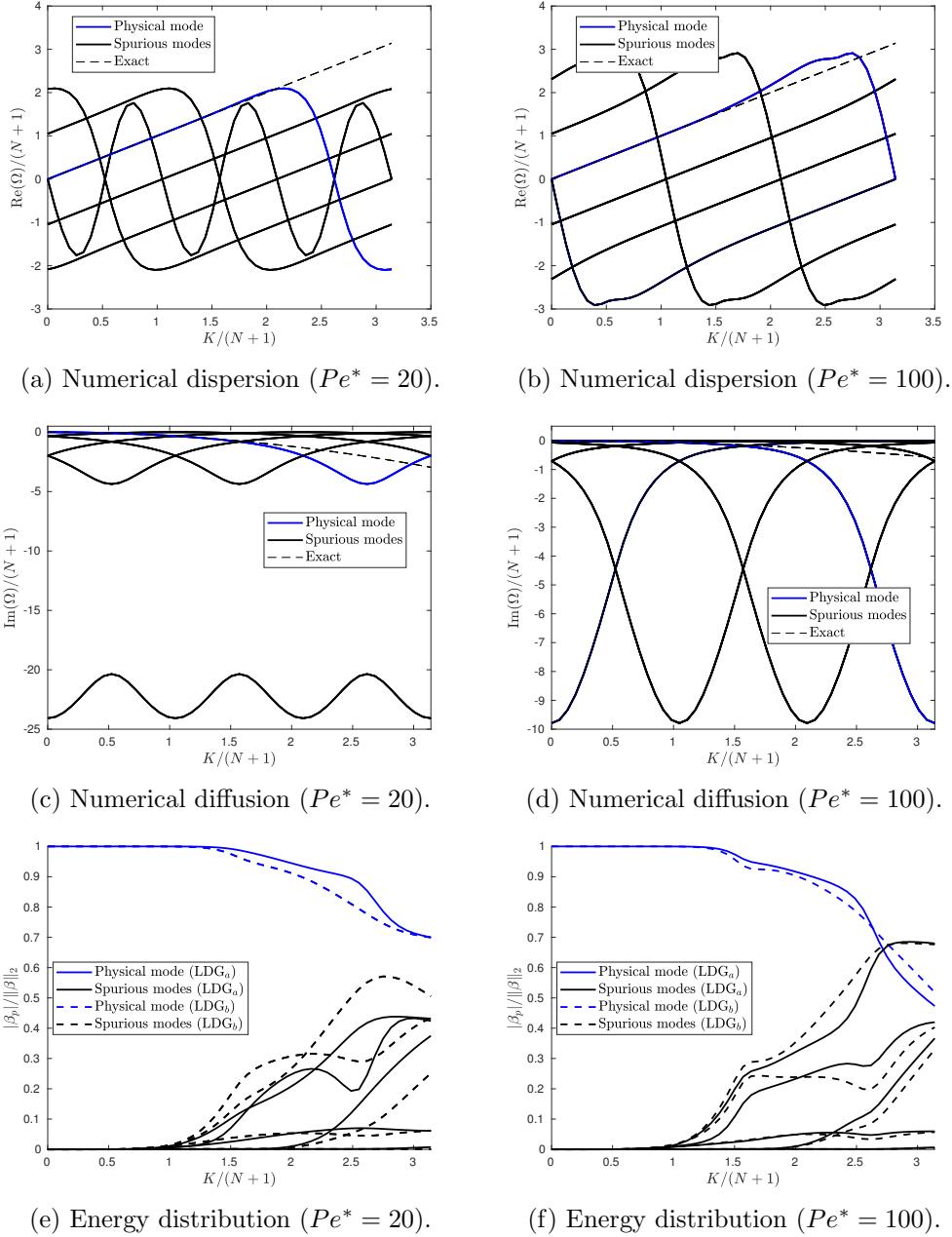


Figure 2.11: Numerical dispersion relation, numerical diffusion and energy distribution of the eigenmodes for the DG scheme of order  $N = 5$  using Legendre-Gauss nodes for  $Pe^* = 20$  (first column) and  $Pe^* = 100$  (second column). Diffusion term discretized by the two alternate variants LDG<sub>a</sub> (solid lines) and LDG<sub>b</sub> (dashed lines).

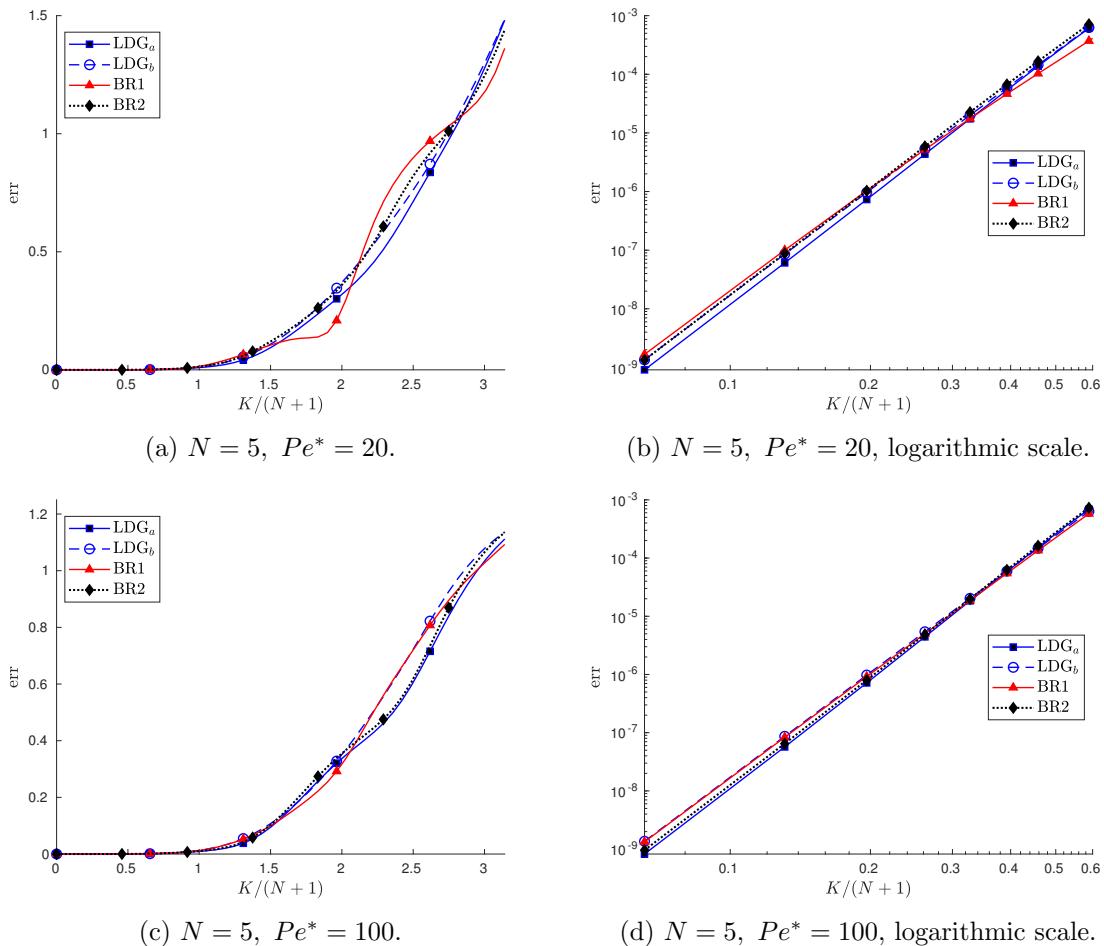


Figure 2.12: Relative error  $\text{err}(t)$  vs. non-dimensional wave number for the  $\text{DG}(N = 5)$  scheme on Legendre-Gauss nodes for  $Pe^* = 20$  (first row) and  $Pe^* = 100$  (second row).

as

$$\|\mathbf{v}\|_{L^2} = \sqrt{\frac{\Delta x}{2} \sum_{j=1}^E \sum_{\nu=1}^{N+1} \omega_\nu v_{j,\nu}^2}, \quad (2.52)$$

where  $\omega_\nu$ ,  $\nu = 1, \dots, N+1$ , denote either the Legendre-Gauss or the Legendre-Gauss-Lobatto weights corresponding to the DG nodal set employed.

### Evolution of a single wave

The purpose of this first test case is to verify the error analysis in Section 2.4.2. We consider initial solutions of the form

$$U_0(x) = \sin(K(x - t)) \quad (2.53)$$

for various sizes of the wave number  $K$ . Tables 2.1 and 2.2 list the  $L^2$ -errors of the numerical solution versus the non-dimensional wave number  $\frac{K}{N+1}$  for the DG schemes for  $N = 1, \dots, 5$  on Legendre-Gauss and Legendre-Gauss-Lobatto nodes, respectively. The simulations are run until final time  $t = 0.05$  and the grid Peclet number is set to  $Pe^* = 20$ , according to the eigenanalysis in Section 2.4.2. For this test case, the DG scheme is combined with both variants of the alternate LDG diffusion flux as well as the BR1 and BR2 diffusion discretization. The study in Section 2.4.2 regarding the dissipation and dispersion properties with respect to the BR2 penalty parameter is continued by choosing  $\eta_e = \frac{N}{N+1}, 2, 3$  for the DG scheme on Legendre-Gauss nodes and  $\eta_e = 2, 3, \frac{3(N+1)}{N}$  for the DG scheme on Legendre-Gauss-Lobatto nodes using the BR2<sub>LGL</sub> implementation with inexact calculation of the BR2 lifting operator through (2.12) as described in Section 2.1. We recall from our analysis in Section 2.3, the BR2<sub>LGL</sub>( $\eta_e = 3\frac{N+1}{N}$ ) scheme is equivalent to BR2<sub>LG</sub>( $\eta_e = 3$ ) and for the DG scheme on Legendre-Gauss-Lobatto nodes, BR2<sub>LGL</sub>( $\eta_e = 1$ ) is equivalent to the BR1 scheme. Regarding the numerical results listed in Tables 2.1 and 2.2, the following observations can be made:

- In almost all set-ups, the LDG<sub>a</sub> variant performs better than LDG<sub>b</sub>.
- In case of Legendre-Gauss nodes, for all  $N$ , there is some range of wave numbers where the BR1 scheme and the BR2 scheme for  $\eta_3 = \frac{N}{N+1}$  beat the LDG<sub>a</sub> discretization and the BR2( $\eta_3 = \frac{N}{N+1}$ ) scheme actually performs best of all schemes.
- On Legendre-Gauss-Lobatto nodes, LDG<sub>b</sub> yields the largest error of all schemes except for the case  $N = 4$ ,  $\frac{K}{N+1} = \frac{4\pi}{5}$ , with similarly large errors of all schemes. Furthermore, increasing the penalty parameter of the BR2 scheme increases the accuracy of the numerical solution on this set of nodes except for the cases  $N = 1$ ,  $\frac{K}{N+1} = \frac{\pi}{3}$  and  $N = 5$ ,  $\frac{K}{N+1} = \frac{\pi}{2}$ .
- We observe consistency of the above results with the eigensolution analysis in Section 2.4.2 by considering Figures 2.6, 2.7, 2.8, 2.9, 2.10 and 2.12.

N	$K/N + 1$	LDG <sub>a</sub>	LDG <sub>b</sub>	BR1	BR2		
					$\eta_e = N/N + 1$	$\eta_e = 2$	$\eta_e = 3$
1	$\pi/10$	3.27e-03	5.97e-03	4.74e-03	4.75e-03	4.61e-03	4.54e-03
	$\pi/5$	1.27e-02	2.20e-02	1.71e-02	1.71e-02	1.72e-02	1.74e-02
	$\pi/2$	1.12e-01	1.22e-01	1.07e-01	9.00e-02	1.16e-01	1.33e-01
	$4\pi/5$	2.32e-01	2.33e-01	1.96e-01	1.86e-01	2.32e-01	2.63e-01
	$9\pi/10$	3.01e-01	3.01e-01	2.59e-01	2.56e-01	3.01e-01	3.31e-01
2	$\pi/10$	1.16e-03	1.32e-03	9.36e-04	7.61e-04	1.19e-03	1.48e-03
	$\pi/5$	7.65e-03	1.17e-02	7.78e-03	7.54e-03	9.19e-03	1.05e-02
	$\pi/2$	6.45e-02	1.74e-01	1.27e-01	1.31e-01	1.18e-01	1.10e-01
	$4\pi/5$	3.75e-01	4.27e-01	3.29e-01	2.95e-01	3.97e-01	4.64e-01
	$9\pi/10$	5.03e-01	5.21e-01	3.98e-01	3.84e-01	5.10e-01	5.90e-01
3	$\pi/10$	1.68e-04	3.82e-04	2.65e-04	2.72e-04	2.48e-04	2.47e-04
	$\pi/5$	4.20e-03	5.38e-03	3.65e-03	3.15e-03	4.53e-03	5.39e-03
	$\pi/2$	9.83e-02	6.93e-02	2.45e-02	2.45e-02	8.18e-02	1.04e-01
	$4\pi/5$	4.85e-01	6.50e-01	5.14e-01	4.90e-01	5.56e-01	6.00e-01
	$9\pi/10$	7.10e-01	7.84e-01	6.02e-01	5.74e-01	7.35e-01	8.08e-01
4	$\pi/10$	5.24e-05	7.22e-05	5.07e-05	4.95e-05	6.06e-05	6.69e-05
	$\pi/5$	7.06e-04	3.41e-04	1.06e-04	1.06e-04	4.84e-04	6.12e-04
	$\pi/2$	1.19e-01	1.41e-01	7.93e-02	6.38e-02	1.25e-01	1.44e-01
	$4\pi/5$	8.33e-01	9.77e-01	8.90e-01	8.60e-01	9.24e-01	9.44e-01
	$9\pi/10$	9.03e-01	9.86e-01	8.34e-01	8.09e-01	9.16e-01	9.53e-01
5	$\pi/10$	1.05e-05	1.50e-05	1.28e-05	1.28e-05	1.41e-05	1.50e-05
	$\pi/5$	6.74e-04	8.80e-04	4.30e-04	2.31e-04	7.11e-04	8.50e-04
	$\pi/2$	1.12e-01	1.84e-01	1.16e-01	9.20e-02	1.47e-01	1.63e-01
	$4\pi/5$	7.42e-01	8.24e-01	9.35e-01	9.74e-01	8.81e-01	8.55e-01
	$9\pi/10$	1.15e+00	1.17e+00	1.12e+00	1.11e+00	1.13e+00	1.14e+00

Table 2.1: Comparison of relative  $L^2$ -errors depending on the non-dimensional wave number for the DG scheme on Legendre-Gauss nodes.

$N$	$K/N + 1$	$\text{LDG}_a$	$\text{LDG}_b$	$\text{BR1}$	BR2		
					$\eta_e = 2$	$\eta_e = 3$	$\eta_e = 3(N+1)/N$
1	$\pi/10$	8.39e-03	1.01e-02	9.31e-03	9.27e-03	9.22e-03	9.09e-03
	$\pi/5$	3.32e-02	3.91e-02	3.62e-02	3.61e-02	3.60e-02	3.55e-02
	$\pi/3$	9.13e-02	1.02e-01	9.59e-02	9.63e-02	9.68e-02	9.86e-02
	$4\pi/5$	3.14e-01	3.15e-01	3.15e-01	3.15e-01	3.15e-01	3.15e-01
	$9\pi/10$	3.34e-01	3.34e-01	3.34e-01	3.34e-01	3.34e-01	3.34e-01
2	$\pi/10$	2.31e-03	2.46e-03	2.39e-03	2.38e-03	2.37e-03	2.35e-03
	$\pi/5$	1.68e-02	1.93e-02	1.81e-02	1.79e-02	1.77e-02	1.75e-02
	$\pi/2$	1.72e-01	2.31e-01	2.03e-01	1.99e-01	1.95e-01	1.89e-01
	$4\pi/5$	5.08e-01	5.42e-01	5.25e-01	5.22e-01	5.20e-01	5.18e-01
	$9\pi/10$	6.14e-01	6.27e-01	6.20e-01	6.19e-01	6.18e-01	6.17e-01
3	$\pi/10$	4.43e-04	5.93e-04	5.25e-04	5.03e-04	4.88e-04	4.76e-04
	$\pi/5$	7.33e-03	8.87e-03	8.15e-03	7.95e-03	7.81e-03	7.72e-03
	$\pi/2$	1.32e-01	1.58e-01	1.49e-01	1.43e-01	1.39e-01	1.36e-01
	$4\pi/5$	6.49e-01	7.34e-01	6.87e-01	6.75e-01	6.67e-01	6.62e-01
	$9\pi/10$	8.11e-01	8.65e-01	8.32e-01	8.25e-01	8.21e-01	8.19e-01
4	$\pi/10$	9.92e-05	1.36e-04	1.20e-04	1.13e-04	1.10e-04	1.09e-04
	$\pi/5$	9.92e-04	1.53e-03	1.37e-03	1.20e-03	1.12e-03	1.09e-03
	$\pi/2$	1.92e-01	2.44e-01	2.20e-01	2.11e-01	2.07e-01	2.07e-01
	$4\pi/5$	1.01e+00	9.98e-01	1.01e+00	1.00e+00	1.00e+00	1.00e+00
	$9\pi/10$	1.12e+00	1.17e+00	1.13e+00	1.12e+00	1.12e+00	1.12e+00
5	$\pi/10$	2.02e-05	3.02e-05	2.49e-05	2.31e-05	2.27e-05	2.26e-05
	$\pi/5$	1.13e-03	1.79e-03	1.41e-03	1.31e-03	1.31e-03	1.31e-03
	$\pi/2$	2.35e-01	2.99e-01	2.53e-01	2.52e-01	2.54e-01	2.55e-01
	$4\pi/5$	1.17e+00	1.23e+00	1.22e+00	1.19e+00	1.18e+00	1.17e+00
	$9\pi/10$	1.53e+00	1.54e+00	1.53e+00	1.52e+00	1.52e+00	1.52e+00

Table 2.2: Comparison of relative  $L^2$ -errors depending on the non-dimensional wave number for the DG scheme on Legendre-Gauss-Lobatto nodes.

### A well-resolved test case

As in Watkins et al. [210] we now consider the solution of the 1D advection-diffusion equation with a well-resolved approximate Gaussian as initial condition. Hereby, the initial condition  $\mathbf{u}(0)$  as well as the exact solution  $\mathbf{u}^{ex}(t)$  can be computed from the analytical solution

$$U(x, t) = \sum_{\mu=-N_{\hat{k}}}^{N_{\hat{k}}} \theta_\mu e^{-d\hat{k}_\mu^2 t} \cos(\hat{k}_\mu(x - t)), \quad (2.54)$$

where  $\theta_\mu$  is the  $\mu$ -th spectral weight given below,  $\hat{k}_\mu = 2\pi\mu/L$  is the  $\mu$ -th wave number associated with the domain length  $L = 20$  and  $N_{\hat{k}}$  is the number of waves used, where  $N_{\hat{k}}$  is chosen the largest positive integer such that  $\hat{k}_{N_{\hat{k}}} \leq (N + 1)\pi/\Delta x$ . The spectral weights  $\theta_\mu$  are defined as

$$\theta_\mu = \frac{e^{-(\sigma\hat{k}_\mu)^2/2}}{\sqrt{2\pi}\sigma \sum_{s=-N_{\hat{k}}}^{N_{\hat{k}}} e^{-(\sigma\hat{k}_s)^2/2}}, \quad -N_{\hat{k}} \leq \mu \leq N_{\hat{k}}, \quad (2.55)$$

where  $\sigma$  is the standard deviation of the Gaussian which dictates its width. For the well-resolved Gaussian, we set  $\sigma = 8/\sqrt{2\pi}$ .

We now compute the numerical solution  $\mathbf{u}(t)$  using the DG scheme for  $N = 1, \dots, 5$  with different diffusion discretizations. Tables 2.3 and 2.4 contain the relative  $L^2$ -errors  $\|\mathbf{u}(t) - \mathbf{u}^{ex}(t)\|_{L^2}/\|\mathbf{u}^{ex}(t)\|_{L^2}$  for the various versions of the DG scheme at output times  $t = 0.1, 1, 10$ . The results in the top rows of Tables 2.3 and 2.4 show that

- regarding the two alternative LDG variants,  $\text{LDG}_a$  yields lower numerical errors in all set-ups.
- for odd polynomial degrees  $N = 1, 3, 5$ , the  $\text{LDG}_a$  diffusion discretization performs best for all investigated output times and both nodal sets except for the case  $N = 1, t = 1$  on Legendre-Gauss-Lobatto nodes. Furthermore, the BR2 schemes with large penalty parameters  $\eta_e = 2, 3$  and  $\eta_e = \frac{3(N+1)}{N}$  beat the BR1 scheme in all odd degree cases, with errors decreasing for increasing  $\eta_e$ , except for the  $\text{DG}(N = 1)$  scheme on Legendre-Gauss-Lobatto nodes and output time  $t = 10$ . In addition, on Legendre-Gauss nodes, the BR2 scheme with small penalty parameter  $\eta_e = \frac{N}{N+1}$  yields larger errors than BR1 for  $N = 3, 5$ .
- for even polynomial degrees  $N = 2, 4$ , the situation above is reversed, i.e. the BR2 scheme for  $\eta_e = \frac{N}{N+1}$  performs best of all diffusion discretizations in case of Legendre-Gauss nodes followed by the BR1 scheme. Furthermore, regarding the BR2 penalty parameter, increasing  $\eta_e$  increases the error of the numerical solution. For the even degree case and Legendre-Gauss-Lobatto nodes, either the  $\text{LDG}_a$  variant or the BR1 scheme yield the lowest numerical error, also depending on the output time  $t$ .

### A low-resolution test case

For this test case, basically the same set-up is used as for the previous test case. However, the standard deviation is reduced to  $\sigma = 1/\sqrt{2\pi}$  to produce a poorly resolved initial condition. The

last five rows of Tables 2.3 and 2.4 contain the relative  $L^2$ -errors at output times  $t = 0.1, 1, 10$  for this low-resolution test case. The results show that

- regarding the two alternative LDG variants,  $\text{LDG}_a$  yields lower numerical errors in all set-ups except for the  $\text{DG}(N = 4)$  scheme on Legendre-Gauss nodes at  $t = 0.1$  where both variants yield almost the same error.
- On Legendre-Gauss-Lobatto nodes, the behavior of the schemes with respect to accuracy corresponds to the error plots in Section 2.4.2 for low output times  $t = 0.1$  (for all polynomial degrees) and  $t = 1$  (for  $N = 2, 4$ ). More precisely, in these cases, the most accurate diffusion scheme is  $\text{LDG}_a$ , followed by the BR2 schemes for  $\eta_e = \frac{3(N+1)}{N}$ ,  $\eta_e = 3$  and  $\eta_e = 2$ , in this order, while the second largest error is given by the BR1 scheme and the largest one by the  $\text{LDG}_b$  approach.
- On Legendre-Gauss nodes, the most accurate approach is  $\text{LDG}_a$  except for the three cases  $N = 1$ ,  $t = 1$ ;  $N = 3$ ,  $t = 10$  and  $N = 4$ ,  $t = 0.1$ . Furthermore, considering the BR1 and BR2 approaches and even polynomial degrees  $N = 2, 4$ , the order of these schemes with respect to accuracy is BR2 for  $\eta_e = 3, 2$ , BR1, and lastly BR2 for  $\eta_e = \frac{N}{N+1}$ .

$N$	$t$	$\text{LDG}_a$	$\text{LDG}_b$	$\text{BR1}$	BR2		
					$\eta_e = N/N + 1$	$\eta_e = 2$	$\eta_e = 3$
Well-resolved test case							
1	0.1	1.20e-03	2.20e-03	1.81e-03	1.82e-03	1.70e-03	1.62e-03
	1	2.20e-03	4.01e-03	3.89e-03	3.92e-03	3.10e-03	2.73e-03
	10	3.42e-03	4.60e-03	4.91e-03	4.96e-03	4.09e-03	3.69e-03
2	0.1	9.31e-05	9.85e-05	7.67e-05	6.05e-05	9.33e-05	1.12e-04
	1	1.38e-04	1.50e-04	1.11e-04	8.76e-05	1.35e-04	1.62e-04
	10	1.23e-04	1.33e-04	9.85e-05	7.77e-05	1.19e-04	1.44e-04
3	0.1	1.96e-06	4.57e-06	4.30e-06	4.38e-06	3.24e-06	2.88e-06
	1	2.98e-06	5.43e-06	6.46e-06	6.59e-06	4.74e-06	4.21e-06
	10	2.64e-06	4.66e-06	5.61e-06	5.72e-06	4.13e-06	3.68e-06
4	0.1	1.20e-07	1.52e-07	9.28e-08	7.22e-08	1.27e-07	1.48e-07
	1	1.30e-07	1.75e-07	9.38e-08	7.16e-08	1.32e-07	1.57e-07
	10	1.08e-07	1.44e-07	7.67e-08	5.77e-08	1.08e-07	1.29e-07
5	0.1	3.12e-09	4.91e-09	6.26e-09	6.26e-09	5.32e-09	5.15e-09
	1	3.21e-09	5.28e-09	7.07e-09	7.24e-09	5.61e-09	5.35e-09
	10	2.51e-09	4.13e-09	5.63e-09	5.75e-09	4.42e-09	4.20e-09
Low-resolution test case							
1	0.1	3.46e-02	5.06e-02	3.93e-02	4.04e-02	4.06e-02	4.14e-02
	1	2.08e-01	2.20e-01	2.08e-01	1.95e-01	2.14e-01	2.25e-01
	10	1.32e-01	1.33e-01	1.40e-01	1.41e-01	1.37e-01	1.35e-01
2	0.1	2.14e-02	5.66e-02	4.61e-02	4.73e-02	3.86e-02	3.41e-02
	1	4.04e-02	4.98e-02	5.67e-02	5.75e-02	4.82e-02	4.45e-02
	10	9.30e-03	9.66e-03	1.12e-02	1.13e-02	1.04e-02	1.02e-02
3	0.1	9.46e-03	1.35e-02	1.14e-02	1.11e-02	1.14e-02	1.19e-02
	1	6.81e-03	7.68e-03	7.24e-03	6.86e-03	7.65e-03	8.15e-03
	10	4.44e-04	5.27e-04	4.46e-04	4.21e-04	4.88e-04	5.31e-04
4	0.1	2.12e-03	2.11e-03	2.99e-03	3.13e-03	2.56e-03	2.47e-03
	1	1.39e-03	1.92e-03	2.28e-03	2.28e-03	1.94e-03	1.89e-03
	10	2.83e-05	4.22e-05	4.94e-05	5.01e-05	4.13e-05	4.05e-05
5	0.1	6.60e-04	1.16e-03	1.27e-03	1.27e-03	1.15e-03	1.14e-03
	1	2.07e-04	2.87e-04	3.19e-04	3.19e-04	2.86e-04	2.94e-04
	10	2.20e-06	3.32e-06	2.40e-06	2.27e-06	2.77e-06	3.00e-06

Table 2.3: Relative  $L^2$ -errors for the well-resolved test case with  $\sigma = 8/\sqrt{2\pi}$  and for the low-resolution test case with  $\sigma = 1/\sqrt{2\pi}$ . The DG schemes use Legendre-Gauss (LG) nodes as well as different diffusion discretizations.

$N$	$t$	LDG <sub>a</sub>	LDG <sub>b</sub>	BR1	BR2		
					$\eta_e = 2$	$\eta_e = 3$	$\eta_e = 3(N + 1)/N$
Well-resolved test case							
1	0.1	3.45e-03	4.19e-03	3.85e-03	3.82e-03	3.78e-03	3.68e-03
	1	1.60e-02	1.90e-02	1.86e-02	1.75e-02	1.66e-02	1.44e-02
	10	5.33e-02	5.42e-02	5.33e-02	5.37e-02	5.41e-02	5.50e-02
2	0.1	2.19e-04	2.36e-04	2.28e-04	2.26e-04	2.25e-04	2.24e-04
	1	6.04e-04	7.65e-04	6.00e-04	6.45e-04	6.87e-04	7.45e-04
	10	5.88e-04	7.16e-04	5.90e-04	6.22e-04	6.53e-04	6.97e-04
3	0.1	8.05e-06	1.03e-05	9.56e-06	8.98e-06	8.64e-06	8.44e-06
	1	1.82e-05	2.46e-05	2.54e-05	2.13e-05	1.95e-05	1.86e-05
	10	1.57e-05	2.09e-05	2.15e-05	1.83e-05	1.68e-05	1.60e-05
4	0.1	2.72e-07	4.01e-07	3.15e-07	3.06e-07	3.08e-07	3.10e-07
	1	4.59e-07	7.35e-07	4.42e-07	4.70e-07	4.96e-07	5.10e-07
	10	3.86e-07	6.08e-07	3.73e-07	3.95e-07	4.16e-07	4.27e-07
5	0.1	9.57e-09	1.39e-08	1.26e-08	1.12e-08	1.08e-08	1.06e-08
	1	1.26e-08	2.14e-08	2.19e-08	1.68e-08	1.53e-08	1.49e-08
	10	9.94e-09	1.68e-08	1.75e-08	1.33e-08	1.21e-08	1.18e-08
Low-resolution test case							
1	0.1	1.10e-01	1.29e-01	1.20e-01	1.19e-01	1.18e-01	1.16e-01
	1	6.00e-01	6.29e-01	6.05e-01	6.11e-01	6.18e-01	6.40e-01
	10	5.07e-01	5.09e-01	4.93e-01	5.06e-01	5.17e-01	5.47e-01
2	0.1	8.04e-02	1.04e-01	9.43e-02	9.10e-02	8.82e-02	8.47e-02
	1	1.01e-01	1.12e-01	1.10e-01	1.05e-01	1.03e-01	1.01e-01
	10	6.57e-02	6.64e-02	7.07e-02	6.69e-02	6.45e-02	6.20e-02
3	0.1	2.35e-02	2.75e-02	2.58e-02	2.51e-02	2.48e-02	2.46e-02
	1	2.12e-02	2.43e-02	2.49e-02	2.26e-02	2.18e-02	2.17e-02
	10	3.86e-03	4.07e-03	3.79e-03	3.84e-03	3.91e-03	3.97e-03
4	0.1	5.59e-03	7.52e-03	6.52e-03	6.03e-03	5.86e-03	5.81e-03
	1	5.23e-03	7.11e-03	6.17e-03	5.59e-03	5.44e-03	5.40e-03
	10	1.94e-04	2.31e-04	2.46e-04	2.14e-04	2.04e-04	2.01e-04
5	0.1	2.00e-03	2.71e-03	2.49e-03	2.29e-03	2.24e-03	2.22e-03
	1	7.05e-04	1.06e-03	1.12e-03	8.55e-04	7.91e-04	7.73e-04
	10	8.45e-06	1.35e-05	8.33e-06	8.54e-06	8.77e-06	8.87e-06

Table 2.4: Relative  $L^2$ -errors for the well-resolved test case with  $\sigma = 8/\sqrt{2\pi}$  and for the low-resolution test case with  $\sigma = 1/\sqrt{2\pi}$ . The DG schemes use Legendre-Gauss-Lobatto (LGL) nodes as well as different diffusion discretizations.

## Chapter 3

# Recent Advances on Time Discretization

### *IMEX Advection-Diffusion Splitting and Positivity Preservation*

In the method of lines approach, once a suitable space discretization has been carried out, such as a classical SBP finite difference scheme for the linear advection equation as in Section 1.1 or a DG scheme applied to hyperbolic conservation laws in cell-wise form as in Section 1.2 , a system of ordinary differential equations (ODEs) is obtained, as can be seen in equations (1.3) and (1.53), respectively. We denote this resulting system of ODEs by

$$\frac{d\mathbf{u}}{dt} = \mathbf{g}(t, \mathbf{u}), \quad (3.1)$$

where  $\mathbf{u}$  now contains all spatial degrees of freedom and  $\mathbf{g}$  denotes the operator corresponding to the spatial discretization. In principle, the above system can then be solved by any numerical scheme designed for the approximate solution of ordinary differential equations and in the previous numerical examples we already made use of such schemes. In the context of the method of lines approach, this process is referred to as time discretization and the chosen numerical scheme is the time integrator. Basically, two classes of time integrators can be distinguished: explicit and implicit ones.

#### Explicit time integration

Many advantages can be listed in favor of explicit time integration schemes. They are easier to implement because they do not require the solution of nonlinear systems and are usually quite robust. Explicit time integrators evaluate the right hand side of the system of ODEs (3.1) by using only already known data as input values for  $\mathbf{u}$ , either at the current time level  $t^n$ , at computed intermediate stages or at previous time levels. Very popular explicit time

integrators are Runge-Kutta (RK) schemes which have the form

$$\begin{aligned}\mathbf{u}^{(i)} &= \mathbf{u}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{g}(t^n + c_j \Delta t, \mathbf{u}^{(j)}) , \quad i = 1, \dots, s, \\ \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t \sum_{i=1}^s b_i \mathbf{g}(t^n + c_i \Delta t, \mathbf{u}^{(i)}) ,\end{aligned}\tag{3.2}$$

where  $\Delta t$  denotes the time step size,  $\mathbf{u}^n$  provides the approximation to  $\mathbf{u}(t)$  at time  $t^n$  and  $\mathbf{u}^{(i)}$  approximates  $\mathbf{u}(t)$  at suitable intermediate stages corresponding to time levels  $t^n + c_i \Delta t$  with  $c_i = \sum_{j=1}^{i-1} a_{ij}$ . The simplest example is the explicit Euler, or forward Euler method

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \mathbf{g}(t^n, \mathbf{u}^n) .$$

As mentioned above, these explicit schemes are easy to implement, also in parallel hardware environment, need a comparatively small amount of CPU time per time step, and may be constructed to have a high order of accuracy. However, their range of stability is limited. For conservation laws describing fluid flow, the allowable time step size depends on the characteristic speed and, in case of a global time step size in the computational domain, on the length scale of the smallest cell. Due to this severe time step restriction in case of locally refined grids, explicit time integrators are sometimes used in a multirate fashion, allowing different time step sizes in different parts of the computational domain, see e.g. the investigations by Osher and Sanders [154], Tang and Warnecke [191], Constantinescu and Sandu [46], Hundsdorfer et al. [83], Schlegel et al. [172], Gassner et al. [64], and Seny et al. [174].

### Implicit time integration

Implicit time integration methods involve the solution of linear or nonlinear equations to determine the numerical solution in the next time step. Thus, the computational effort per time step is larger than for explicit schemes and parallel computing is more difficult to realize. However, implicit schemes may be preferred over explicit ones if the time step constraints ensuring stability are much more restrictive than those necessary to achieve the desired accuracy of the numerical solution. Then, the implicit scheme can take larger time steps and thus be more efficient. Problems of this kind which are more efficiently solved by an implicit scheme are called stiff.

In case of space-discretized partial differential equations stiffness may be introduced by particular terms such as diffusion terms or specific reaction terms. In addition, stiffness may be caused by only a small subset of the degrees of freedom in the space discretization, for instance in case of grid refinement.

In an implicit RK scheme, the stage values are now implicitly determined by

$$\mathbf{u}^{(i)} = \mathbf{u}^n + \Delta t \sum_{j=1}^s a_{ij} \mathbf{g}(t^n + c_j \Delta t, \mathbf{u}^{(j)}) ,$$

whereas the computation of  $\mathbf{u}^{n+1}$  from the stage values  $\mathbf{u}^{(i)}$  is of the same form as in (3.2).

A very popular subclass of implicit RK schemes are diagonally implicit (DIRK) schemes with

$$\mathbf{u}^{(i)} = \mathbf{u}^n + \Delta t \sum_{j=1}^i a_{ij} \mathbf{g}(t^n + c_j \Delta t, \mathbf{u}^{(j)}) .$$

For these schemes, the size of the non-linear systems is reduced since each stage is equivalent, in terms of computational effort, to the implicit Euler scheme  $\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \mathbf{g}(t^n + \Delta t, \mathbf{u}^{n+1})$ . The resulting nonlinear systems may be linearized by Newton's method leading to a sequence of linear systems to be solved. The solution of these usually large linear systems again involves iterative schemes such as preconditioned Krylov subspace methods.

### Organization of this chapter

Advanced time integration schemes often consist in combinations of known methods or modify existing schemes in order to increase efficiency or stability or with the purpose of transferring additional properties of the analytical solution to the numerically computed approximation.

Regarding the first approach, the first three sections of this chapter deal with combinations of explicit and implicit time integration methods increasing efficiency of fluid flow simulations while maintaining stability. In this context, Section 3.1 reviews general implicit-explicit approaches based on different splittings between implicitly and explicitly discretized terms or degrees of freedom. Section 3.2 deals with the specific case of advection-diffusion splitting where only the diffusion terms are treated implicitly. Finally, Section 3.3 considers the stability of DG schemes using implicit-explicit time integration based on advection-diffusion splitting.

Modifications of known methods are usually demanded in order to achieve additional properties related to the specific application. Preserving positivity or non-negativity of certain physical quantities often is an important additional requirement regarding numerical methods in the context of fluid simulations since disrespecting this property often leads to stability issues. In this regard, Section 3.4 deals with positivity preserving schemes for semi-discretizations of partial differential equations in production-destruction form.

## 3.1 Implicit-Explicit (IMEX) approaches for fluid flow equations

Considering the numerical simulation of fluid flow, in particular using the compressible Navier-Stokes equations, explicit time integration may become inefficient due to numerical stiffness caused for example by boundary layers, acoustic waves or due to the presence of viscous terms. Closely related, stiffness caused by viscous terms is also a well-known drawback to the use of purely explicit time integration schemes for other convection-diffusion type problems in particular when using high order discretizations in space such as spectral methods. However, purely implicit time discretization applied to spatial discretizations of fluid flow equations requires the solution of large non-linear systems of equations and may thus be computationally expensive as well. Therefore, hybrid time integration schemes such as implicit-explicit (IMEX) methods have frequently been considered.

IMEX time integration methods for problems with multiple time scales such as fluid flow or convection-diffusion type problems are generally based either on combinations of linear multistep methods or on partitioned Runge-Kutta schemes, each applied to a partitioned system of ordinary differential equations of the form

$$\frac{d\mathbf{u}}{dt} = \mathbf{g}_1(t, \mathbf{u}) + \mathbf{g}_2(t, \mathbf{u}). \quad (3.3)$$

Hereby, different time integration schemes, i.e. an explicit and an implicit one, are applied to the components  $\mathbf{g}_1$  and  $\mathbf{g}_2$  of the split right-hand side.

In order to obtain (3.3), an adequate splitting of the right-hand side of the semi-discrete system has to be determined either a-priori or adaptively during the numerical simulation. Of course, any suitable approach needs to take into account the kind of stiffness of the given problem. A natural choice based on stiffness due to viscous terms is *advection-diffusion splitting*, whereby the advective terms are discretized explicitly while the diffusive terms are treated implicitly, see e.g. [8, 201, 7, 30]. This choice of splitting covers the fact that explicit time discretization of the viscous fluxes results in an increasingly severe grid-dependent time step restriction since the time step  $\Delta t$  scales with the grid length scale  $\Delta x$  as  $\Delta t = \mathcal{O}((\Delta x)^2)$  whereas for pure advection problems, explicit time integration is stable under a more moderate time step restriction of  $\Delta t = \mathcal{O}(\Delta x)$ . In fact, this is the reason why the simulation of convection-diffusion type problems using purely explicit time stepping is often not advisable. In addition, implicit time stepping only applied to the viscous terms often reduces the computational effort in comparison to a fully implicit approach since even though many applications contain non-linear convection terms, the diffusion terms are often linear, only demanding the iterative solution of linear algebraic systems which are positive definite, symmetric and sparse. Last but not least, the non-linear systems resulting from potentially non-linear diffusion terms are generally more efficient to solve than those arising from non-linear advection terms. The stability of IMEX Runge-Kutta schemes based on advection-diffusion splitting will be dealt with in more detail in the following Section 3.2 and in Section 3.3 in case of DG discretizations.

Advection-diffusion splitting deposes of the severe time step restrictions due to the effects of diffusion. However, with this approach, all degrees of freedom of the complete computational domain contribute to the implicit part of the time integration scheme. Nonetheless, for many applications, the properties of the fluid flow and thus the cell sizes vary considerably within the computational domain, e.g. if boundary layers are to be accurately resolved. This leads to so-called *geometry-induced stiffness* where the restrictive conditions on the time step size in case of explicit schemes are due to only a moderate number of small elements of the computational grid. For such problems, a reasonable splitting of the form (3.3) needs to distinguish between parts of the grid to be discretized explicitly and part of it to be discretized implicitly resulting in *domain-based splitting*. In addition, this approach primarily reduces the memory requirements regarding the implicit part of the discretization since fewer degrees of freedom contribute to the implicit terms and the Jacobian matrices and preconditioners to be stored are smaller. A successful application of IMEX-RK schemes using domain-based splitting in combination with the DG space discretization is given by Kanevsky et al. in [95] using an a-priori splitting into implicit and explicit regions of the computational domain based on the cell sizes. In [175], based on a finite volume discretization in space, Shoeybi et al. consider a related approach referred to as *row-splitted* IMEX scheme where the splitting of

the degrees of freedom into explicit and implicit sets does not directly rely on the cell sizes but calculates the sizes of the Gershgorin circles corresponding to each row of the global Jacobi matrix of the right-hand side of (3.1) in order to estimate the contribution of the respective row to the spectral radius. Based on a user-specified maximum admissible spectral radius of the explicit part, the rows of the semi-discrete system (3.1) are then either assigned to  $\mathbf{g}_1$  or to  $\mathbf{g}_2$  in (3.3). This approach considers that both the computational grid and the properties of the fluid flow itself may induce stiffness. Furthermore, the splitting procedure operates in time allowing to adapt the decomposition at every time step during the simulation. In [198], Vermeire and Nadarajah extend the row-splitted IMEX scheme in order to combine it with a higher order discretization in space based on the flux reconstruction approach. Hereby, after estimating their influence on the spectral radius via Gershgorin circles as in [175], the degrees of freedom of an entire element are moved either to the explicit or the implicit set resulting in a conservative IMEX splitting. While the domain-based splitting approaches in [95, 175, 198] utilize IMEX-RK schemes as specific time integrators, Straub et al. [185, 184] construct new domain-based IMEX time integrators based on exponential integrators. Hereby, the splitting of the right-hand side of (3.1) into two terms  $\mathbf{g}_1, \mathbf{g}_2$  to be discretized explicitly and implicitly, respectively, is realized by setting  $\mathbf{g}_1(t, \mathbf{u}) = \mathbf{L}(t)\mathbf{u}$  and  $\mathbf{g}_2(t, \mathbf{u}) = \mathbf{g}(t, u) - \mathbf{L}(t)\mathbf{u} = \mathbf{N}(t, \mathbf{u})$ . In order to achieve a domain-based splitting, the linear operator  $\mathbf{L}$  by construction only differs from the identity for the implicit subset of the given degrees of freedom and can thus be implemented using matrices  $\tilde{\mathbf{L}}(t)$  of smaller dimension. Time integration of the given ODE

$$\frac{d\mathbf{u}}{dt} = \mathbf{L}(t)\mathbf{u} + \mathbf{N}(t, \mathbf{u})$$

using exponential integrators then requires to approximate matrix exponentials as well as related functions called  $\varphi$ -functions of the matrices  $\tilde{\mathbf{L}}(t)$ . In this context, Straub et al. [185, 184] employ EPIRK and sEPIRK schemes developed by Tokman et al. [195, 194, 164] which incorporate adaptive Krylov subspace projections to increase efficiency of their exponential integrators. These and similar recent approaches improving the design of such  $\varphi$  solvers have rendered exponential integrators serious competitors to widely used classical implicit RK schemes. Also in a domain-based IMEX setting, numerical experiments in [183] simulating two-dimensional inviscid fluid flow in a nozzle have shown increased efficiency of IMEX-type schemes based on exponential integrators beating classical IMEX-RK schemes by factors up to 2.5 for third order time integration and up to 6 for first order time integration.

## 3.2 IMEX-RK schemes for advection-diffusion splitting

As explained in the previous Section 3.1, in case of advection-diffusion splitting, the right-hand side of the semi-discrete system (3.3) is split such that the semi-discrete advection terms are discretized explicitly and the semi-discrete diffusion terms are discretized implicitly.

Although IMEX linear multistep methods have been applied in the context of advection-diffusion splitting, e.g. in [8], Ascher et al. argue in [7] that multirate schemes applied in this context may lead to undesirable time step restrictions unless diffusion is the dominant phenomenon and BDF based schemes are chosen. Consequently, they develop IMEX Runge-Kutta schemes for convection-diffusion type problems with superior stability regions covering a larger parameter range by combining DIRK schemes with explicit RK methods.

While the splitting of advection and diffusion terms alleviates the severe time step scaling for the diffusion terms, a CFL-type time step restriction of the form  $\Delta t = \mathcal{O}(\Delta x)$  may still have to be fulfilled due to the explicit time discretization of the advection terms. However, certain IMEX schemes using advection-diffusion splitting possess stronger stability properties. In fact, Calvo et al. [30] showed that some IMEX-RK schemes in [7] require to reduce the time step under grid refinement, when a linear advection-diffusion equation is solved, while others do not. The latter authors then design IMEX methods which guarantee grid-independent time step restrictions of the form  $\Delta t = \mathcal{O}(d/a^2)$  for Fourier spatial discretization of linear advection-diffusion equations, where  $a$  and  $d$  denote the advection and diffusion coefficient, respectively. Hereby, the gist of these specifically designed IMEX schemes is that the implicitly discretized diffusion terms stabilize the explicit discretization of the advection terms, and consequently, the coupled scheme has better stability properties for advection-diffusion as compared to its explicit part applied to the pure advection problem.

A related approach is taken in [169, 173] where the authors design unconditionally stable IMEX linear multistep schemes where both the IMEX splitting and the employed multistep scheme fulfill specific properties. This confirms once more that choosing arbitrarily large time steps without losing stability may indeed be possible even though some parts of the problem are discretized explicitly.

Returning to IMEX-RK schemes, we follow the notation in [7, 30] and consider an  $(s+1)$ -stage explicit Runge-Kutta method  $(\mathbf{A}^{(1)}, \mathbf{b}^{(1)}, \mathbf{c}^{(1)})$  coupled with an implicit  $s$ -stage DIRK scheme and  $(\mathbf{A}^{(2)}, \mathbf{b}^{(2)}, \mathbf{c}^{(2)})$  where the abscissae  $\mathbf{c}^{(1)}, \mathbf{c}^{(2)}$  of the explicit and implicit scheme, respectively, fulfill  $\mathbf{c}^{(1)} = \begin{pmatrix} 0 \\ \mathbf{c}^{(2)} \end{pmatrix}$ . The DIRK scheme is then formally recast into an  $(s+1)$ -stage scheme as well by padding the first row and first column with zeros. The first stage of the coupled scheme is thus explicit and we obtain the Butcher array

$$\begin{array}{c|ccccc|ccccc} 0 & 0 & & & & & 0 & & & \\ c_2 & a_{21}^{(1)} & 0 & & & & 0 & a_{22}^{(2)} & & \\ c_3 & a_{31}^{(1)} & a_{32}^{(1)} & 0 & & & 0 & a_{32}^{(2)} & a_{33}^{(2)} & \\ \vdots & \vdots & \vdots & \ddots & \ddots & & \vdots & \vdots & \ddots & \ddots \\ c_{s+1} & a_{s+1,1}^{(1)} & a_{s+1,2}^{(1)} & \cdots & a_{s+1,s}^{(1)} & 0 & 0 & a_{s+1,2}^{(2)} & \cdots & a_{s+1,s}^{(2)} & a_{s+1,s+1}^{(2)} \\ \hline & b_1^{(1)} & b_2^{(1)} & b_2^{(1)} & \cdots & b_{s+1}^{(1)} & 0 & b_2^{(2)} & b_3^{(2)} & \cdots & b_{s+1}^{(2)} \end{array}$$

Applied to the system of ODEs (3.3), the above IMEX-RK scheme has the form

$$\begin{aligned} \mathbf{y}^{(1)} &= \mathbf{y}^n \\ \mathbf{y}^{(i)} &= \mathbf{y}^n + \Delta t \sum_{j=1}^{s+1} a_{ij}^{(1)} \mathbf{g}_1(t^{n,j}, \mathbf{y}^{(j)}) + \Delta t \sum_{j=1}^{s+1} a_{ij}^{(2)} \mathbf{g}_2(t^{n,j}, \mathbf{y}^{(j)}), \quad i = 2, \dots, s+1, \\ \mathbf{y}^{n+1} &= \mathbf{y}^n + \Delta t \sum_{j=1}^{s+1} b_j^{(1)} \mathbf{g}_1(t^{n,j}, \mathbf{y}^{(j)}) + \Delta t \sum_{j=1}^{s+1} b_j^{(2)} \mathbf{g}_2(t^{n,j}, \mathbf{y}^{(j)}), \end{aligned} \tag{3.4}$$

where  $\mathbf{y}^n, \mathbf{y}^{n+1}$  denote the approximations at times  $t^n$  and  $t^{n+1} = t^n + \Delta t$  and  $\mathbf{y}^{(i)}$  the

intermediate stage values corresponding to the intermediate times  $t^{n,i}$  which are given by  $t^{n,i} = t^n + c_i \Delta t$  with  $c_i = \sum_{j=1}^s a_{ij}^{(1)} = \sum_{j=1}^s a_{ij}^{(2)}$ .

Since the classical Dahlquist test equation  $u'(t) = \lambda u(t)$  is not sufficient to determine the stability of IMEX schemes or other coupled time integration methods, a more suitable test equation needs to be found. In case of IMEX splitting by advection and diffusion terms, considering finite difference approximations provides insight in order to derive more suitable test equations.

We recapitulate the linear advection-diffusion equation

$$\frac{\partial}{\partial t} U(x, t) + a \frac{\partial}{\partial x} U(x, t) = d \frac{\partial^2}{\partial x^2} U(x, t), \quad (x, t) \in Q = \Omega \times (0, T), \quad \Omega = (x_\alpha, x_\beta), \quad (3.5)$$

already introduced in Section 2.4.1, with diffusion coefficient  $d > 0$  and advective velocity  $a > 0$ , supplemented by the periodic initial condition  $U(x, 0) = U_0(x)$  in  $L^2(\Omega)$  and periodic boundary conditions. In the case of space discretization by finite difference methods, we may derive the stability of specific IMEX schemes based on advection-diffusion splitting by considering the eigenvalues of circulant matrices.

In fact, discretizing (3.5) on the domain  $\Omega = (0, 1)$  by second order central finite difference schemes both for the advection and the diffusion terms and setting periodic boundary conditions, we obtain the semi-discretization

$$u'_j(t) = \left( \frac{d}{(\Delta x)^2} + \frac{a}{2\Delta x} \right) u_{j-1}(t) - \frac{2d}{(\Delta x)^2} u_j(t) + \left( \frac{d}{(\Delta x)^2} - \frac{a}{2\Delta x} \right) u_{j+1}(t),$$

where the values  $u_j(t) \approx U(x_j, t)$  denote pointwise approximations of the exact solution at  $m$  equidistant grid points with  $x_j = j\Delta x$ ,  $j = 1, \dots, m$  and  $\Delta x = \frac{1}{m}$ . In matrix-vector formulation, we obtain a system of linear ODEs of the form

$$\mathbf{u}'(t) = \mathbf{G}\mathbf{u}(t) \quad (3.6)$$

with the circulant matrix  $\mathbf{G}$  which decomposes into two circulant matrices corresponding to the advection terms on the one hand and to the diffusion terms on the other hand, i.e.

$$\mathbf{G} = \mathbf{G}_a + \mathbf{G}_d = \frac{a}{2\Delta x} \begin{pmatrix} 0 & -1 & & & 1 \\ 1 & 0 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & -1 \\ -1 & & & 1 & 0 \end{pmatrix} + \frac{d}{(\Delta x)^2} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix}.$$

Since circulant matrices are simultaneously diagonalizable, the system (3.6) decouples into  $m$  scalar linear ODEs. By computing the eigenvalues of  $\mathbf{G}_a$  and  $\mathbf{G}_d$ , see e.g. [84], these equations take the form

$$u'(t) = - \left( \frac{ia}{\Delta x} \sin(2\pi k \Delta x) + \frac{4d}{(\Delta x)^2} \sin^2(\pi k \Delta x) \right) u(t), \quad k = 1, \dots, m. \quad (3.7)$$

In addition, due to the simultaneous diagonalizability of the advection part and the diffusion part of  $\mathbf{G}$ , applying a partitioned RK method (3.4) to the system (3.6) using

$$\mathbf{g}_1(t, \mathbf{y}) = \mathbf{G}_a \mathbf{y}, \quad \mathbf{g}_2(t, \mathbf{y}) = \mathbf{G}_d \mathbf{y}$$

may also be decoupled into scalar expressions involving (3.7). Hence, the behavior of the IMEX-RK scheme (3.4) applied to the scalar ODEs (3.7) solely determines its behavior with respect to the original linear system of ODEs (3.6). Scalar equations of the type

$$u'(t) = \mu u(t) + i\lambda u(t), \quad \lambda, \mu \in \mathbb{R}, \mu \leq 0 \quad (3.8)$$

therefore often serve as test functions to determine the stability of IMEX-RK schemes applied to advection-diffusion equations using advection-diffusion splitting. Rather than simply desiring the stability region of the IMEX-RK scheme to be as large as possible in  $\{(\lambda, \mu) \in \mathbb{R}^2 \mid \mu \leq 0\}$ , a more specific stability condition may be derived from (3.7). In fact, the equations (3.7) show a dependence between the corresponding eigenvalues  $i\lambda = -\frac{ia}{\Delta x} \sin(2\pi k \Delta x)$  of  $\mathbf{G}_a$  and  $\mu = -\frac{4d}{(\Delta x)^2} \sin^2(\pi k \Delta x)$  of  $\mathbf{G}_d$ , particularly under grid refinement. Taking into account the time step length  $\Delta t$ , we have

**Lemma 3.1.** *Let  $z_1 = \lambda \Delta t = -\frac{ia \Delta t}{\Delta x} \sin(2\omega)$  and  $z_2 = \mu \Delta t = -\frac{4d \Delta t}{(\Delta x)^2} \sin^2(\omega)$ , with  $\omega \in [0, \pi]$ . Choosing  $\Delta t$  such that  $\Delta t \leq \alpha^{-1} \frac{d}{a^2}$  for a fixed constant  $\alpha \in \mathbb{R}^+$ , the estimate*

$$|z_2| \geq \alpha z_1^2$$

holds for any value of  $\Delta x > 0$ .

*Proof.* Using the assumption on  $\Delta t$  and the equality  $\sin(2\omega) = 2 \sin \omega \cos \omega$ , we have

$$z_1^2 = \frac{a^2 (\Delta t)^2}{(\Delta x)^2} \sin^2(2\omega) \leq \alpha^{-1} \frac{d \Delta t}{(\Delta x)^2} \sin^2(2\omega) \leq \alpha^{-1} \frac{4d \Delta t}{(\Delta x)^2} \sin^2(\omega) = \alpha^{-1} |z_2|,$$

which proves the assertion.  $\square$

Applying the IMEX-RK scheme (3.4) to the scalar equation (3.8) using  $g_1(t, y) = i\lambda y$ ,  $g_2(t, y) = \mu y$  now yields  $y^{n+1} = R(z_1, z_2)y^n$  with the stability function  $R$  given by

$$R(z_1, z_2) = \frac{\det(\mathbf{I} - iz_1 \mathbf{A}^{(1)} - z_2 \mathbf{A}^{(2)} + iz_1 \mathbf{1}(\mathbf{b}^{(1)})^T + z_2 \mathbf{1}(\mathbf{b}^{(2)})^T)}{\det(\mathbf{I} - iz_1 \mathbf{A}^{(1)} - z_2 \mathbf{A}^{(2)})}.$$

Due to the estimate provided by Lemma 3.1, it is desirable to devise IMEX-RK schemes with the property that there is a constant  $\alpha \in \mathbb{R}^+$  such that the stability region

$$\mathcal{S} = \{(z_1, z_2) \in \mathbb{R}^2 \mid |R(z_1, z_2)| \leq 1\} \quad (3.9)$$

of the IMEX-RK method contains the region  $\{(z_1, z_2) \in \mathbb{R}^2 \mid z_2 \leq -\alpha z_1^2\}$  below the parabola given by  $z_2 = -\alpha z_1^2$ . For the special case  $\mathbf{b}^{(1)} = \mathbf{b}^{(2)}$ , necessary condition on the IMEX-RK coefficients to fulfill this parabola property were identified by Calvo et al in [30]. For general IMEX-RK schemes of the form (3.4), the arguments in their paper easily transfer and we obtain the necessary conditions

$$a_{s+1,j}^{(2)} = b_j^{(2)}, \quad 1 \leq j \leq s+1, \quad (3.10)$$

$$a_{s+1,1}^{(1)} = b_1^{(1)}. \quad (3.11)$$

While in previous works considering IMEX advection-diffusion splitting, finite difference or Fourier-type spatial discretizations have been considered, Wang et al. [205] were the first to study grid-independent  $L^2$ -stability in the context of spatial discretization by a specific discontinuous Galerkin method. Under a time step restriction of the form  $\Delta t = \mathcal{O}(d/a^2)$ , they proved  $L^2$ -stability of the local discontinuous Galerkin (LDG) scheme [44] scheme applied to linear advection-diffusion equations in one space dimension and extended their results to non-linear problems in [204] and to the multi-dimensional case in [208]. Furthermore, in [59], Fu and Shu prove grid-independent  $L^2$ -stability for the discretization of the diffusion terms by the embedded discontinuous Galerkin method. The question arises if this favorable property is also inherent to more general DG diffusion schemes, e.g. if it applies to the widely used schemes developed by Bassi and Rebay in [13, 15, 14] or to the more classical DG approaches by Arnold et al. [5] and Baumann and Oden [17]. Before answering this question for DG diffusion discretization by the  $(\sigma, \mu)$ -family in Section 3.3, we review the identification of suitable IMEX-RK schemes for advection-diffusion splitting in the following Section 3.2.

### Examples of IMEX-RK schemes

A few classical low-order IMEX-RK schemes and their stability properties for advection-diffusion equations will be discussed in the following. When constructing suitable IMEX-RK schemes of a desired convergence order, we may use the known order conditions on the coefficients of more general partitioned Runge-Kutta schemes [75, 90]. However, for low order schemes, a direct Taylor expansion is more constructive. Following this approach shows that first order IMEX-RK schemes of the form (3.4) only need to fulfill the conditions

$$\sum_j^{s+1} b_j^{(1)} = \sum_j^{s+1} b_j^{(2)} = 1, \quad (3.12)$$

while second order schemes in addition to fulfilling (3.12) satisfy the conditions

$$\sum_j^{s+1} b_j^{(1)} c_j = \sum_j^{s+1} b_j^{(2)} c_j = \frac{1}{2}. \quad (3.13)$$

A classical first order IMEX-RK scheme results from combining the explicit and implicit Euler schemes in the form

$$\mathbf{y}^{n+1} = \mathbf{y}^n + \Delta t \mathbf{g}_1(t, \mathbf{y}^n) + \Delta t \mathbf{g}_2(t, \mathbf{y}^{n+1}) \quad (3.14)$$

resulting in the combined Butcher array

$$\begin{array}{c|cc|cc} 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 1 \\ \hline & 1 & 0 & 0 & 1 \end{array} . \quad (3.15)$$

which fulfills the conditions on the coefficients given in (3.10) and (3.11). The stability function of the above IMEX Euler scheme is given by

$$R(z_1, z_2) = \frac{1 + iz_1}{1 - z_2}$$

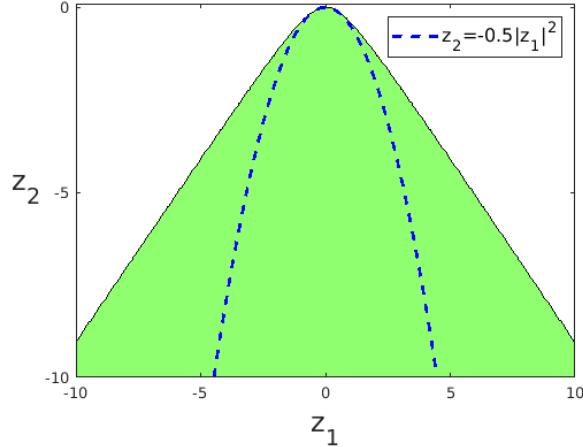


Figure 3.1: IMEX stability region of the scheme (3.15).

and the stability region as defined in (3.9) is plotted in Figure 3.1. Obviously, the region under the parabola defined by  $z_2 = -\frac{1}{2}|z_1|^2$  is completely contained within the stability region of the IMEX Euler scheme. This yields grid-independent stability of this IMEX method when applied to the semi-discrete linear advection-diffusion equation (3.6).

A second order IMEX-RK scheme may be obtained by a combination of the explicit and implicit trapezoidal rule yielding

$$\begin{aligned} \mathbf{y}^* &= \mathbf{y}^n + \Delta t \left( \mathbf{g}_1(\mathbf{y}^n) + \frac{1}{2} (\mathbf{g}_2(\mathbf{y}^n) + \mathbf{g}_2(\mathbf{y}^*)) \right) \\ \mathbf{y}^{n+1} &= \mathbf{y}^n + \frac{1}{2} \Delta t (\mathbf{g}(\mathbf{y}^n) + \mathbf{g}(\mathbf{y}^*)) \end{aligned}$$

with the combined Butcher array

$$\begin{array}{c|cc|cc} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 & 1/2 & 1/2 \end{array} \quad (3.16)$$

and the stability function for IMEX advection-diffusion splitting given by

$$R(z_1, z_2) = 1 - \frac{(iz_1 + 2)(iz_1 + z_2)}{z_2 - 2}.$$

This scheme does not fulfill the condition (3.11). Accordingly, the stability region of this scheme does not fulfill the parabola property as can be seen in Figure 3.2.

In [8], Ascher et al considered a second order IMEX scheme incorporating a stiffly accurate SDIRK scheme. This scheme is defined by the Butcher array

$$\begin{array}{c|ccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \gamma & \gamma & 0 & 0 & 0 & \gamma & 0 \\ 1 & \delta & 1-\delta & 0 & 0 & 1-\gamma & \gamma \\ \hline & \delta & 1-\delta & 0 & 0 & 1-\gamma & \gamma \end{array} \quad (3.17)$$

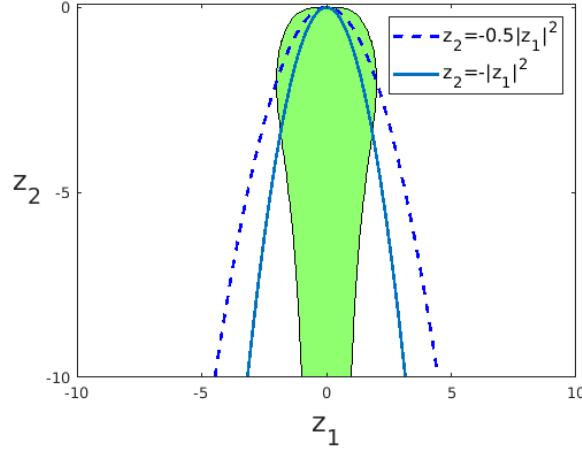


Figure 3.2: IMEX stability region of the scheme (3.16)

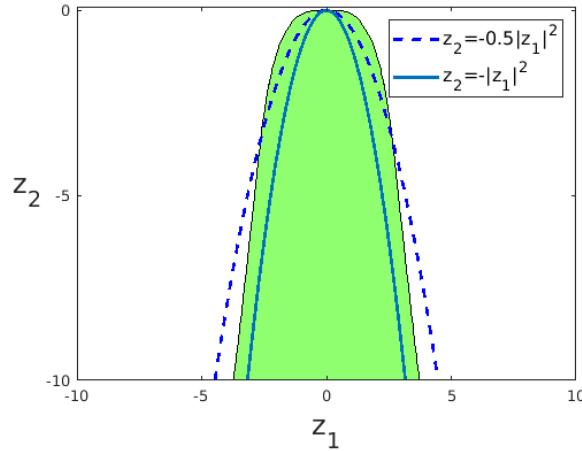


Figure 3.3: IMEX stability region of the scheme (3.17).

with  $\gamma = 1 - \frac{\sqrt{2}}{2}$  and  $\delta = 1 - \frac{1}{2\gamma}$ . Obviously, the conditions (3.10) and (3.11) are fulfilled by the coefficients of this scheme. Furthermore, a plot of the IMEX stability region given in Figure 3.3 indicates that the parabola property is satisfied for the constant  $\alpha = 1$ .

### 3.3 $L^2$ -stability analysis of IMEX- $(\sigma, \mu)$ DG schemes for advection-diffusion

In this section, we review the fully discrete  $L^2$ -stability analysis carried out in [152] for linear advection-diffusion problems in one space dimension which are discretized in space by the DG scheme based on the  $(\sigma, \mu)$ -family of diffusion discretizations already discussed in Section 2.2 and discretized in time by implicit-explicit (IMEX) Runge-Kutta schemes as described in Section 3.2. Hereby, following the approach of IMEX advection-diffusion splitting, advection

terms are discretized explicitly in time while diffusion terms are solved implicitly. The investigation of this approach builds upon the previous results by Wang, Shu and Zhang [205]. Here, the LDG scheme for the diffusive terms used in [205] is replaced by the  $(\sigma, \mu)$ -family. As shown in Section 2.2, this family contains many well-known DG diffusion discretizations as well as some newer ones. While it naturally includes the original DG diffusion discretizations in [5, 17], it contains both the BR1 and BR2 scheme, if the integrals in the DG variational form are numerical solved by Legendre-Gauss-Lobatto quadrature, as well as a symmetrized form of LDG.

From a theoretical analysis, conditions on the given parameters  $\sigma$  and  $\mu$  are derived which guarantee  $L^2$ -stability for time steps  $\Delta t = \mathcal{O}(d/a^2)$ , where  $a$  and  $d$  denote the advection and diffusion coefficient, respectively, i.e. the allowable time step size does not decrease under grid refinement. In the spirit of [205], this is referred to as unconditional  $L^2$ -stability. It turns out that these parameter restrictions are neither fulfilled by the simple BR1 approach nor by the Baumann-Oden (BO) scheme [17]. In addition, corresponding numerical experiments for various members of the  $(\sigma, \mu)$ -family show that the BR1 scheme and the BO approach do not allow for a grid-independent time step choice contrary to those members of the  $(\sigma, \mu)$ -family fulfilling the conditions presented in this work. In fact, both the BR2 scheme (2.11) with penalty constant  $\eta_e > 1$  using the BR2<sub>LGL</sub> implementation (2.12) for the calculation of the lifting operator and the more recent  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme [111, 109] fulfill the presented conditions and yield an unconditionally  $L^2$ -stable IMEX-DG scheme in the sense of [205].

In the following, we present the corresponding theoretical  $L^2$ -stability analysis for the two IMEX time integration schemes of first and second order given by the Butcher arrays (3.15) and (3.17), respectively, which were also used in [205]. IMEX time integration is combined with the DG scheme of arbitrary order in space using  $(\sigma, \mu)$  diffusion fluxes. Numerical results verifying the conducted theoretical analysis are afterwards presented in Section 3.3.3.

## Preliminaries

We consider the space discretization of the linear advection-diffusion equation (3.5) by the DG scheme. Hereby, the computational domain  $\Omega$  is again partitioned into cells  $I_j$ , where presently only uniform grids are considered. Basis functions and test functions used to define the DG scheme are taken from the finite element space

$$V_h = \{v \in L^2(\Omega) \mid v|_{I_j} \in \mathcal{P}_N(I_j) \forall j = 1, \dots, E\},$$

as in Section 2.1.

As in Section 2.4.1, with a slightly modified notation for the discretization of advection terms, the semi-discrete DG scheme for (3.5) is defined as the solution  $u(t) \in V_h$  of the variational formulation

$$(u_t, v)_j = a\mathcal{H}_j(u, v) + d\mathcal{L}_j(u, v), \quad \forall v \in V_h, \quad (3.18)$$

with  $(\cdot, \cdot)_j$  denoting the usual inner product in  $L^2(I_j)$ . Hereby, the advection terms are discretized by the upwind numerical flux resulting in

$$\mathcal{H}_j(u, v) = (u, v_x)_j - u_{j+1}^- v_{j+1}^- + u_j^- v_j^+, \quad (3.19)$$

while the diffusion terms are discretized by the  $(\sigma, \mu)$ -family of diffusion fluxes introduced in Section 2.2, which is specified by the operator  $\mathcal{L}_j$  defined in (2.15).

The following preliminary estimates regarding the global advection operator  $\mathcal{H}$  defined by

$$\mathcal{H}(u, v) = \sum_{j=1}^E \mathcal{H}_j(u, v)$$

may be taken from the stability analysis of IMEX-LDG schemes in [217, 205], where we recall that the  $L^2$ -norm  $\|\cdot\|$  and the jump semi-norm  $[[\cdot]]$  on  $V_h$  are given by

$$\|v\|^2 = (v, v) = \sum_{j=1}^E (v, v)_j, \quad [[v]]^2 = \sum_{j=1}^E [v]_j^2,$$

respectively.

**Lemma 3.2.** *The global upwind operator  $\mathcal{H}$  fulfills the following properties.*

1. *For any  $u, v \in V_h$ , we have*

$$\mathcal{H}(u, u) = -\frac{1}{2}[[u]]^2, \quad (3.20)$$

$$\mathcal{H}(u, v) = -\frac{1}{2}[[v]]^2 - \mathcal{H}(v - u, v). \quad (3.21)$$

2. *For any  $u, v \in V_h$  and for any constant  $C > 0$ , we have the estimate*

$$|a\mathcal{H}(v - u, v)| \leq \frac{d}{C} \left( \|v_x\|^2 + \frac{\nu}{\Delta x} [[v]]^2 \right) + \frac{a^2 C}{2d} \|v - u\|^2, \quad (3.22)$$

where  $a, d$  are the advection and diffusion coefficients, respectively, and  $\nu$  denotes an inverse constant depending on the polynomial degree  $N$  used to define  $V_h$ .

**Remark 3.3.** The values of  $\nu$  in (3.22) for the polynomial degrees  $N = 1, 2, 3$  may be taken from [218] and are given in Table 3.1.

*Proof of Lemma 3.2.* Since  $\mathcal{H}_j(u, v)$  may be rewritten as

$$\mathcal{H}_j(u, v) = -(u_x, v)_j - [u]_j v_j^+,$$

we directly obtain (3.20) via

$$\mathcal{H}(u, u) = \frac{1}{2} \left( - \sum_{j=1}^E [u]_j u_j^+ + \sum_{j=1}^E u_j^- [u]_j \right) = -\frac{1}{2}[[u]]^2.$$

Furthermore, using both (3.20) and the linearity of  $\mathcal{H}(\cdot, \cdot)$  immediately yields (3.21). In addition, from [205], Lemma 2.2, we obtain the estimate

$$|\mathcal{H}(v - u, v)| \leq \left( \|v_x\| + \sqrt{\nu \Delta x^{-1}} [[v]] \right) \|v - u\|, \quad (3.23)$$

with the inverse constant  $\nu$  depending on the polynomial degree  $N$  as given in [218] for  $N = 1, 2, 3$ . From Young's inequality applied separately to the terms  $\|v_x\| \|v - u\|$  and  $\sqrt{\nu \Delta x^{-1}} [[v]] \|v - u\|$  in (3.23), equation (3.22) now follows.  $\square$

$N$	1	2	3
$\nu$	6	12	20

Table 3.1: Values of the inverse constant  $\nu$ .

Furthermore, Lemma 2.1 and Theorem 2.2 in Section 2.2 provide several useful properties regarding symmetry and dissipativity of the  $(\sigma, \mu)$ -family of diffusion schemes. These properties will be needed to study  $L^2$ -stability of the second order IMEX scheme in Section 3.2. Moreover, the condition on  $\sigma$  and  $\mu$  given in Lemma 2.2 in order to achieve dissipativity of  $\mathcal{L}$  shows that we need to demand  $\frac{(1-\sigma)^2}{4\omega_1} \leq \mu$  in order to obtain semi-discrete  $L^2$ -stability, i.e. energy stability, of the spatial diffusion discretization alone, neglecting the potential presence of advection terms.

In case of advection-diffusion equations, discretizing the advection terms by the upwind scheme introduces additional dissipation into the semi-discrete scheme, hence  $L^2$ -stability also follows for the semi-discrete advection-diffusion equation under the above condition. For the fully-discrete schemes analyzed in Sect. 3.2, stability in case of the additional presence of advection terms will demand a slightly stricter condition of  $\frac{(1-\sigma)^2}{4\omega_1} < \mu$  to allow for a grid-independent time step choice. In particular, in case of a stable diffusion discretization only fulfilling  $\frac{(1-\sigma)^2}{4\omega_1} = \mu$ , such as the BR1 scheme, the implicit time discretization of the diffusion term does not provide enough dissipation to counteract the explicitly discretized advection term in case of large time steps.

### 3.3.1 Stability analysis with respect to the first order IMEX scheme

Applying the first order IMEX scheme (3.15) to the semi-discrete DG equation (3.18) yields the fully discrete IMEX-DG scheme

$$(u^{n+1}, v)_j = (u^n, v)_j + \Delta t a \mathcal{H}_j(u^n, v) + \Delta t d \mathcal{L}_j(u^{n+1}, v), \quad \forall v \in V_h. \quad (3.24)$$

Setting  $v = u^{n+1}$  and summing up over all cells, we obtain

$$\begin{aligned} (u^{n+1}, u^{n+1}) &= (u^n, u^{n+1}) + \Delta t a \mathcal{H}(u^n, u^{n+1}) \\ &\quad - \Delta t d \left( (u_x^{n+1}, u_x^{n+1}) + (1-\sigma) \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j + \frac{\mu}{\Delta x} [[u^{n+1}]]^2 \right). \end{aligned} \quad (3.25)$$

Using

$$\begin{aligned} (u^{n+1} - u^n, u^{n+1}) &= \frac{1}{2} (u^{n+1}, u^{n+1}) + \frac{1}{2} (u^{n+1} - u^n, u^{n+1} - u^n) - \frac{1}{2} (u^n, u^n) \\ &= \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) + \frac{1}{2} \|u^{n+1} - u^n\|^2 \end{aligned}$$

together with the relations (3.21) and (3.25), we have

$$\begin{aligned} & \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) \\ &= -\frac{1}{2} \|u^{n+1} - u^n\|^2 - \frac{\Delta t a}{2} [[u^{n+1}]]^2 - \Delta t a \mathcal{H}(u^{n+1} - u^n, u^{n+1}) \\ &\quad - \Delta t d \left( (u_x^{n+1}, u_x^{n+1}) + (1 - \sigma) \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j + \frac{\mu}{\Delta x} [[u^{n+1}]]^2 \right). \end{aligned}$$

The estimate (3.22) then yields

$$\begin{aligned} & \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) \\ &\leq \left( \frac{\Delta t a^2 C}{2d} - \frac{1}{2} \right) \|u^{n+1} - u^n\|^2 + \frac{\Delta t d}{C} \left( \|u_x^{n+1}\|^2 + \frac{\nu}{\Delta x} [[u^{n+1}]]^2 \right) \\ &\quad - \Delta t d \left( \|u_x^{n+1}\|^2 + (1 - \sigma) \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j + \frac{\mu}{\Delta x} [[u^{n+1}]]^2 \right), \end{aligned}$$

and thus

$$\begin{aligned} & \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) \leq \left( \frac{\Delta t a^2 C}{2d} - \frac{1}{2} \right) \|u^{n+1} - u^n\|^2 \\ &\quad - \Delta t d \frac{C - 1}{C} \|u_x^{n+1}\|^2 - \Delta t d \frac{\mu - \frac{\nu}{C}}{\Delta x} [[u^{n+1}]]^2 \\ &\quad - \Delta t d (1 - \sigma) \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j. \end{aligned} \tag{3.26}$$

Obviously, in order to obtain a non-positive contribution of the term containing  $\|u_x^{n+1}\|$ , we need to restrict the constant  $C$  to  $C \geq 1$ .

If  $\sigma \neq 1$ , further restricting to  $C > 1$  and setting  $\tilde{C} = \frac{C-1}{C} \frac{1}{|\sigma-1|}$  in (2.21), we obtain

$$\left| (1 - \sigma) \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j \right| \leq \frac{C - 1}{C} \|u_x^{n+1}\|^2 + \frac{C}{C - 1} \frac{(1 - \sigma)^2}{4\Delta x \omega_1} [[u^{n+1}]]^2.$$

Inserted into (3.26), this yields

$$\begin{aligned} & \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) \leq \left( \frac{\Delta t a^2 C}{2d} - \frac{1}{2} \right) \|u^{n+1} - u^n\|^2 \\ &\quad + \Delta t d \left( \frac{C}{C - 1} \frac{(1 - \sigma)^2}{4\Delta x \omega_1} - \frac{\mu - \frac{\nu}{C}}{\Delta x} \right) [[u^{n+1}]]^2. \end{aligned} \tag{3.27}$$

Hence (3.27) shows that  $L^2$ -stability can now be guaranteed if the conditions

$$\frac{\Delta t a^2 C}{2d} \leq \frac{1}{2} \tag{3.28}$$

$$\frac{C}{C - 1} \frac{(1 - \sigma)^2}{4\omega_1} \leq \mu - \frac{\nu}{C} \tag{3.29}$$

are fulfilled, where the constant  $C > 1$  may be suitably chosen to fulfill the above inequalities. However, via (3.29), the range of admissible  $(\sigma, \mu)$ -schemes is restricted as well.

In Theorem 2.2 we demand  $\frac{(1-\sigma)^2}{4\omega_1} \leq \mu$  in order to obtain  $L^2$ -stability of the diffusion discretization itself. However, due to the occurrence of the explicitly discretized advective terms, here the parameters need to fulfill  $\frac{(1-\sigma)^2}{4\omega_1} < \mu$ . More precisely, if we set

$$\tilde{\mu} := \frac{(1-\sigma)^2}{4\omega_1},$$

the condition (3.29) can be rewritten as

$$\tilde{\mu}C^2 + (C-1)\nu \leq \mu C(C-1) \Leftrightarrow C(\nu + \mu) \leq (\mu - \tilde{\mu})C^2 + \nu.$$

This can be fulfilled if and only if

$$\tilde{\mu} < \mu, \quad (3.30)$$

e.g. by choosing

$$C \geq \frac{\nu + \mu}{\mu - \tilde{\mu}}. \quad (3.31)$$

Under the restriction (3.30), we may hence achieve  $L^2$ -stability for the fully discrete IMEX-DG scheme. In fact, once the parameters of the  $(\sigma, \mu)$ -family are chosen such that they meet the above condition (3.30), the constant  $C$  may be adjusted using (3.31) in order to fulfill the second condition (3.29). With  $C$  determined, the time step  $\Delta t$  is then restricted according to the first condition (3.28) in order to achieve  $L^2$ -stability.

If  $\sigma = 1$ , the last term on the right-hand side of (3.26) vanishes. Condition (3.29) is then replaced by  $\mu \geq \frac{\nu}{C}$ . Since  $\nu \neq 0$ , this condition can only be fulfilled if  $\mu > 0$ . Analogous to the case of  $\sigma \neq 1$ , the time step needs to be restricted by (3.28) where  $C$  is set to  $C = \max\{\frac{\nu}{\mu}, 1\}$ .

Both cases may be merged by demanding  $\tilde{\mu} := \frac{(1-\sigma)^2}{4\omega_1} < \mu$  for the  $(\sigma, \mu)$ -scheme. Summarizing the above findings we have the following

**Theorem 3.4.** *Let  $N \in \mathbb{N}_0$  denote the polynomial degree of the DG approximation space  $V_h$  defined in (2.2). Let  $\sigma, \mu \in \mathbb{R}$  fulfill the condition*

$$\tilde{\mu} := \frac{(1-\sigma)^2}{4\omega_1} < \mu,$$

*where  $\omega_1$  denotes the first weight of the Legendre-Gauss-Lobatto quadrature with  $N+1$  nodes. Then, the IMEX-DG scheme (3.24) with the  $(\sigma, \mu)$ -diffusion operator  $\mathcal{L}_j$  defined in (2.15) fulfills  $\|u^{n+1}\| \leq \|u^n\|$  if the time step is bounded by*

$$\Delta t \leq \frac{d}{a^2 C},$$

*where the constant  $C$  is set to  $C = \max\{\frac{\nu}{\mu}, 1\}$ , if  $\sigma = 1$ , and to  $C = \frac{\nu+\mu}{\mu-\tilde{\mu}}$ , if  $\sigma \neq 1$ .*

As we have shown in the above analysis, if  $\tilde{\mu} = \mu$  in case of  $\sigma \neq 1$ , the condition (3.29) cannot be fulfilled for any  $C > 1$ . Even if setting  $C = 1$  in (3.22) and obtaining a different estimate,

$$|a\mathcal{H}(u^{n+1} - u^n, u^{n+1})| \leq d \left( \|u_x^{n+1}\|^2 + \frac{\nu}{\Delta x} [[u^{n+1}]]^2 \right) + \frac{a^2}{2d} \|u^{n+1} - u^n\|^2,$$

this yields, in combination with (3.26), the estimate

$$\begin{aligned} \frac{1}{2} (\|u^{n+1}\|^2 - \|u^n\|^2) &\leq \left( \frac{\Delta t a^2}{2d} - \frac{1}{2} \right) \|u^{n+1} - u^n\|^2 - \Delta t d \cdot 0 \cdot \|u_x^{n+1}\|^2 \\ &\quad - \Delta t d \left( 2\sqrt{\omega_1 \mu} \sum_{j=1}^E \{u_x^{n+1}\}_j [u^{n+1}]_j + \frac{\mu - \nu}{\Delta x} [[u^{n+1}]]^2 \right), \end{aligned} \quad (3.32)$$

where  $4\omega_1\mu = (1 - \sigma)^2$  is used. Hence, we have

$$2\sqrt{\omega_1 \mu} \sum_{j=1}^E |\{u_x\}_j [u]_j| \leq \sum_{j=1}^E \left( \Delta x \omega_1 \{u_x^2\}_j + \frac{\mu}{\Delta x} [u]_j^2 \right). \quad (3.33)$$

Therefore, the left-hand side term in (3.33), also occurring on the right-hand side of (3.32) cannot be bounded any more by the vanishing contribution of  $\|u_x^{n+1}\|^2$ . In particular, we may therefore assume that the BR1 scheme implemented on Legendre-Gauss-Lobatto nodes, with  $\sigma = -1, \mu = \frac{1}{\omega_1}$  and  $\tilde{\mu} = \mu$ , will not admit a grid independent time step restriction guaranteeing  $L^2$ -stability when combined with IMEX time integration. Neither does the Baumann-Oden method ( $\sigma = 1, \mu = 0$ ) fulfill the conditions on  $\sigma$  and  $\mu$  given in Theorem 3.4. This less favorable behavior of the BR1 and BO schemes in comparison to the LDG scheme and other members of the  $(\sigma, \mu)$ -family such as BR2 will be shown experimentally for the second-order IMEX-DG scheme in Section 3.3.3.

### 3.3.2 Stability analysis with respect to the second order IMEX scheme

Applying the second order IMEX scheme (3.17) to the semi-discrete DG equation (3.18) yields the fully discrete IMEX-DG scheme

$$(u^{(n,1)}, v)_j = (u^n, v)_j + \Delta t \left( \gamma a\mathcal{H}_j(u^n, v) + \gamma d\mathcal{L}_j(u^{(n,1)}, v) \right), \quad (3.34)$$

$$\begin{aligned} (u^{n+1}, v)_j &= (u^n, v)_j + \Delta t \left( \delta a\mathcal{H}_j(u^n, v) + (1 - \delta)a\mathcal{H}_j(u^{(n,1)}, v) \right) \\ &\quad + \Delta t \left( (1 - \gamma)d\mathcal{L}_j(u^{(n,1)}, v) + \gamma d\mathcal{L}_j(u^{n+1}, v) \right), \end{aligned} \quad (3.35)$$

for any function  $v \in V_h$ .

The  $L^2$ -stability analysis for the  $(\sigma, \mu)$ -family applied to the diffusion terms may only partially follow the analysis for the LDG scheme in [205] since the  $(\sigma, \mu)$ -family contains non-symmetric schemes. In addition, the analysis differs where the LDG scheme makes use of the auxiliary variable  $q \approx u_x$ .

As in [205], from (3.34) and (3.35), we get

$$(u^{(n,1)} - u^n, v)_j = \Delta t \left( \gamma a \mathcal{H}_j(u^n, v) + \gamma d \mathcal{L}_j(u^{(n,1)}, v) \right), \quad (3.36)$$

$$\begin{aligned} (u^{n+1} - u^{(n,1)}, v)_j &= \Delta t \left( (\delta - \gamma) a \mathcal{H}_j(u^n, v) + (1 - \delta) a \mathcal{H}_j(u^{(n,1)}, v) \right) \\ &\quad + \Delta t \left( (1 - 2\gamma) d \mathcal{L}_j(u^{(n,1)}, v) + \gamma d \mathcal{L}_j(u^{n+1}, v) \right). \end{aligned} \quad (3.37)$$

We then set  $v = u^{(n,1)}$  in (3.36) and  $v = \frac{1+\theta}{2}u^{n+1} + \frac{1-\theta}{2}u^{(n,1)}$  in (3.37), where  $\theta \in (0, 1)$  is a free parameter chosen to allow for a larger range of admissible non-symmetric members within the  $(\sigma, \mu)$ -family. This parameter will also enter into the time step constraint. Adding both equations (3.36) and (3.37) up over all elements, we now obtain

$$\frac{1}{2} \left( \|u^{n+1}\|^2 - \|u^n\|^2 + \theta \|u^{n+1} - u^{(n,1)}\|^2 + \|u^{(n,1)} - u^n\|^2 \right) = \Delta t (R_1 + R_2), \quad (3.38)$$

with

$$\begin{aligned} R_1 &= \gamma a \mathcal{H} \left( u^n, u^{(n,1)} \right) + \frac{\delta - \gamma}{2} a \mathcal{H} \left( u^n, (1 + \theta)u^{n+1} + (1 - \theta)u^{(n,1)} \right) \\ &\quad + \frac{1 - \delta}{2} a \mathcal{H} \left( u^{(n,1)}, (1 + \theta)u^{n+1} + (1 - \theta)u^{(n,1)} \right) \\ &= \left( \frac{(1 - \theta)\delta + (1 + \theta)\gamma}{2} \right) a \left( \mathcal{H} \left( u^{(n,1)}, u^{(n,1)} \right) - \mathcal{H} \left( u^{(n,1)} - u^n, u^{(n,1)} \right) \right) \\ &\quad + \frac{\theta + 1}{2} (1 - \gamma) a \mathcal{H} \left( u^{n+1}, u^{n+1} \right) + \frac{1 - \theta}{2} (1 - \delta) a \mathcal{H} \left( u^{(n,1)}, u^{(n,1)} \right) \\ &\quad - \frac{\theta + 1}{2} (\delta - \gamma) a \mathcal{H} \left( u^{(n,1)} - u^n, u^{n+1} \right) - \frac{\theta + 1}{2} (1 - \gamma) a \mathcal{H} \left( u^{n+1} - u^{(n,1)}, u^{n+1} \right), \end{aligned}$$

and

$$\begin{aligned} R_2 &= d \left( \gamma \mathcal{L}(u^{(n,1)}, u^{(n,1)}) + \frac{1 - 2\gamma}{2} \mathcal{L}(u^{(n,1)}, (1 + \theta)u^{n+1} + (1 - \theta)u^{(n,1)}) \right. \\ &\quad \left. + \frac{\gamma}{2} \mathcal{L}(u^{n+1}, (1 + \theta)u^{n+1} + (1 - \theta)u^{(n,1)}) \right) \\ &= d \left( \tilde{\beta}_1 \mathcal{L}(u^{(n,1)}, u^{(n,1)}) + \tilde{\beta}_2 \mathcal{L}(u^{(n,1)}, u^{n+1}) + \tilde{\beta}_3 \mathcal{L}(u^{n+1}, u^{(n,1)}) + \tilde{\beta}_4 \mathcal{L}(u^{n+1}, u^{n+1}) \right), \end{aligned}$$

where

$$\tilde{\beta}_1 = \theta\gamma + \frac{1 - \theta}{2}, \quad \tilde{\beta}_2 = \frac{(1 - 2\gamma)(1 + \theta)}{2}, \quad \tilde{\beta}_3 = \frac{\gamma(1 - \theta)}{2}, \quad \tilde{\beta}_4 = \frac{\gamma(1 + \theta)}{2}.$$

Using (3.20), we have

$$\begin{aligned} R_1 &= -a \frac{1 - \theta + (1 + \theta)\gamma}{4} [[u^{(n,1)}]]^2 - a \frac{(\theta + 1)(1 - \gamma)}{4} [[u^{n+1}]]^2 \\ &\quad - a \frac{1 - \theta + (1 + \theta)\gamma}{2} \mathcal{H} \left( u^{(n,1)} - u^n, u^{(n,1)} \right) \\ &\quad + \frac{\theta + 1}{2} \left( a \mathcal{H} \left( u^{(n,1)} - u^n, u^{n+1} \right) - (1 - \gamma) a \mathcal{H} \left( u^{n+1} - u^{(n,1)}, u^{n+1} \right) \right), \end{aligned}$$

since  $\delta - \gamma = -1$ . Different from the corresponding analysis for the LDG scheme in [205], we cannot make use of an auxiliary variable  $q$  to bound the terms of type  $\mathcal{H}(u - v, w)$ . Instead, we use the estimate (3.22) to obtain

$$\begin{aligned} R_1 &\leq \alpha_1 \left( \frac{d}{C_1} \left( \|u_x^{(n,1)}\|^2 + \frac{\nu}{\Delta x} [[u^{(n,1)}]]^2 \right) + \frac{a^2 C_1}{2d} \|u^{(n,1)} - u^n\|^2 \right) \\ &\quad + \alpha_2 \left( \frac{d}{C_2} \left( \|u_x^{n+1}\|^2 + \frac{\nu}{\Delta x} [[u^{n+1}]]^2 \right) + \frac{a^2 C_2}{2d} \|u^{(n,1)} - u^n\|^2 \right) \\ &\quad + \alpha_3 \left( \frac{d}{C_3} \left( \|u_x^{n+1}\|^2 + \frac{\nu}{\Delta x} [[u^{n+1}]]^2 \right) + \frac{a^2 C_3}{2d} \|u^{n+1} - u^{(n,1)}\|^2 \right), \end{aligned}$$

with

$$\alpha_1 = \frac{1 - \theta + (1 + \theta)\gamma}{2}, \quad \alpha_2 = \frac{\theta + 1}{2}, \quad \alpha_3 = \frac{\theta + 1}{2}(1 - \gamma). \quad (3.39)$$

The non-symmetric contributions of  $\mathcal{L}(u^{(n,1)}, u^{n+1})$  and  $\mathcal{L}(u^{n+1}, u^{(n,1)})$  in the definition of  $R_2$  may be dealt with by setting  $\mathbf{u} = (u^{(n,1)}, u^{n+1})^T$  in Theorem 2.2 and splitting into symmetric and anti-symmetric part. Since the symmetric matrix

$$B = \begin{pmatrix} \frac{2\tilde{\beta}_1}{3} & \frac{\tilde{\beta}_2 + \tilde{\beta}_3}{2} \\ \frac{\tilde{\beta}_2 + \tilde{\beta}_3}{2} & \frac{2\tilde{\beta}_4}{3} \end{pmatrix} = \begin{pmatrix} \frac{\theta(2\gamma-1)+1}{3} & \frac{(1-2\gamma)(1+\theta)+\gamma(1-\theta)}{4} \\ \frac{(1-2\gamma)(1+\theta)+\gamma(1-\theta)}{4} & \frac{\gamma(1+\theta)}{3} \end{pmatrix}$$

is positive definite for the parameter range  $\theta \in [0, 0.7]$ , Theorem 2.2 yields

$$\underline{\mathcal{L}}(\mathbf{u}, B\mathbf{u}) \leq 0,$$

for these values of  $\theta$ . This bounds  $R_2$  by

$$\begin{aligned} R_2 &\leq d \left( \frac{\tilde{\beta}_1}{3} \mathcal{L}(u^{(n,1)}, u^{(n,1)}) + \frac{\tilde{\beta}_2 - \tilde{\beta}_3}{2} (\mathcal{L}(u^{(n,1)}, u^{n+1}) - \mathcal{L}(u^{n+1}, u^{(n,1)})) \right. \\ &\quad \left. + \frac{\tilde{\beta}_4}{3} \mathcal{L}(u^{n+1}, u^{n+1}) \right) \end{aligned}$$

for  $\theta \in [0, 0.7]$ .

At this point, we sum up the estimates of the right-hand side terms  $R_1$  and  $R_2$ . Hereby, we make use of the representation (2.20) of  $\mathcal{L}$  and rename the parameters to

$$\begin{aligned} \beta_1 &= \frac{\tilde{\beta}_1}{3} = \frac{\theta(2\gamma-1)+1}{6}, \\ \beta_2 &= \frac{\tilde{\beta}_2 - \tilde{\beta}_3}{2} = \frac{1 + \theta - \gamma(3 + \theta)}{4}, \\ \beta_3 &= \frac{\tilde{\beta}_4}{3} = \frac{\gamma(1 + \theta)}{6}. \end{aligned} \quad (3.40)$$

This yields

$$\begin{aligned}
R_1 + R_2 &\leq d \left( \left( \frac{\alpha_1}{C_1} - \beta_1 \right) \|u_x^{(n,1)}\|^2 + \frac{\frac{\alpha_1\nu}{C_1} - \beta_1\mu}{\Delta x} [[u^{(n,1)}]]^2 \right) \\
&+ d \left( \left( \frac{\alpha_2}{C_2} + \frac{\alpha_3}{C_3} - \beta_3 \right) \|u_x^{n+1}\|^2 + \frac{\frac{\alpha_2\nu}{C_2} + \frac{\alpha_3\nu}{C_3} - \beta_3\mu}{\Delta x} [[u^{n+1}]]^2 \right) \\
&+ d |1 - \sigma| \left( \beta_1 \left| \sum_{j=1}^E \{u_x^{(n,1)}\}_{j+1} [u^{(n,1)}]_{j+1} \right| + \beta_3 \left| \sum_{j=1}^E \{u_x^{n+1}\}_{j+1} [u^{n+1}]_{j+1} \right| \right) \\
&+ d |1 + \sigma| \beta_2 \left( \left| \sum_{j=1}^E \{u_x^{(n,1)}\}_{j+1} [u^{n+1}]_{j+1} \right| + \left| \sum_{j=1}^E \{u_x^{n+1}\}_{j+1} [u^{(n,1)}]_{j+1} \right| \right) \\
&+ \frac{a^2}{2d} \left( (\alpha_1 C_1 + \alpha_2 C_2) \|u^{(n,1)} - u^n\|^2 + \alpha_3 C_3 \|u^{n+1} - u^{(n,1)}\|^2 \right)
\end{aligned}$$

To simplify the analysis, we now choose the constants  $C_1, C_2, C_3$  such that the equation

$$\frac{\alpha_1}{C_1} = \frac{\beta_1}{\beta_3} \left( \frac{\alpha_2}{C_2} + \frac{\alpha_3}{C_3} \right)$$

is fulfilled, e.g. by setting  $C_2 = \frac{2\beta_1\alpha_2}{\beta_3\alpha_1} C_1$ ,  $C_3 = \frac{2\beta_1\alpha_3}{\beta_3\alpha_1} C_1$ .

Thereby, setting  $C = \frac{\beta_1 C_1}{\alpha_1}$ , we obtain the simplified estimate

$$\begin{aligned}
R_1 + R_2 &\leq -d\beta_1 \frac{C-1}{C} \left( \|u_x^{(n,1)}\|^2 + \frac{\beta_3}{\beta_1} \|u_x^{n+1}\|^2 \right) \\
&- d\beta_1 \frac{\mu - \frac{\nu}{C}}{\Delta x} \left( [[u^{(n,1)}]]^2 + \frac{\beta_3}{\beta_1} [[u^{n+1}]]^2 \right) \\
&+ L_1 + L_2 + L_3 + L_4 \\
&+ \frac{a^2}{2d} \left( (\alpha_1 C_1 + \alpha_2 C_2) \|u^{(n,1)} - u^n\|^2 + \alpha_3 C_3 \|u^{n+1} - u^{(n,1)}\|^2 \right),
\end{aligned}$$

with

$$\begin{aligned}
L_1 &= d |1 - \sigma| \beta_1 \left| \sum_{j=1}^E \left\{ u_x^{(n,1)} \right\}_j [u^{(n,1)}]_j \right| \\
&\leq d |1 - \sigma| \beta_1 \left( \tilde{C}_1 \|u_x^{(n,1)}\|^2 + \frac{1}{4\tilde{C}_1 \Delta x \omega_1} [[u^{(n,1)}]]^2 \right), \\
L_2 &= d |1 - \sigma| \beta_3 \left| \sum_{j=1}^E \left\{ u_x^{n+1} \right\}_j [u^{n+1}]_j \right| \\
&\leq d |1 - \sigma| \beta_3 \left( \tilde{C}_2 \|u_x^{n+1}\|^2 + \frac{1}{4\tilde{C}_2 \Delta x \omega_1} [[u^{n+1}]]^2 \right), \\
L_3 &= d |1 + \sigma| \beta_2 \left| \sum_{j=1}^E \left\{ u_x^{(n,1)} \right\}_j [u^{n+1}]_j \right| \\
&\leq d |1 + \sigma| \beta_2 \left( \tilde{C}_3 \|u_x^{(n,1)}\|^2 + \frac{1}{4\tilde{C}_3 \Delta x \omega_1} [[u^{n+1}]]^2 \right), \\
L_3 &= d |1 + \sigma| \beta_2 \left| \sum_{j=1}^E \left\{ u_x^{n+1} \right\}_j [u^{(n,1)}]_j \right| \\
&\leq d |1 + \sigma| \beta_2 \left( \tilde{C}_4 \|u_x^{n+1}\|^2 + \frac{1}{4\tilde{C}_4 \Delta x \omega_1} [[u^{(n,1)}]]^2 \right),
\end{aligned}$$

according to the estimate (2.21).

Setting the constants  $\tilde{C}_2 = \tilde{C}_1$ ,  $\tilde{C}_3 = \sqrt{\frac{\beta_1}{\beta_3}} \tilde{C}_1$ ,  $\tilde{C}_4 = \sqrt{\frac{\beta_3}{\beta_1}} \tilde{C}_1$  and defining

$$\tilde{C}_\sigma = |1 - \sigma| + |1 + \sigma| \frac{\beta_2}{\sqrt{\beta_1 \beta_3}},$$

we then have

$$\begin{aligned}
R_1 + R_2 &\leq d \left( \tilde{C}_1 \tilde{C}_\sigma - \frac{C - 1}{C} \right) \left( \beta_1 \|u_x^{(n,1)}\|^2 + \beta_3 \|u_x^{n+1}\|^2 \right) \\
&\quad d \frac{\frac{\nu}{C} + \tilde{C}_\sigma (4\tilde{C}_1 \omega_1)^{-1} - \mu}{\Delta x} \left( \beta_1 [[u^{(n,1)}]]^2 + \beta_3 [[u^{n+1}]]^2 \right) \\
&\quad + \frac{a^2}{2d} \left( (\alpha_1 C_1 + \alpha_2 C_2) \|u^{(n,1)} - u^n\|^2 + \alpha_3 C_3 \|u^{n+1} - u^{(n,1)}\|^2 \right).
\end{aligned} \tag{3.41}$$

Obviously, we need  $C > 1$ , i.e.  $C_1 > \frac{\alpha_1}{\beta_1}$  in order to obtain a non-positive factor in front of  $\|u_x\|^2$  in the above estimation of  $R_1 + R_2$ .

The last term on the right-hand side of (3.41) may be bounded by the corresponding left-hand side terms in (3.38), yielding the grid-independent conditions

$$\Delta t \frac{a^2}{d} (\alpha_1 C_1 + \alpha_2 C_2) \leq 1, \quad \Delta t \frac{a^2}{d} \alpha_3 C_3 \leq \theta. \tag{3.42}$$

Furthermore, the precise conditions on the parameters  $\sigma$  and  $\mu$  may be obtained in a manner similar to the analysis for the first order IMEX scheme.

Setting  $\tilde{C}_1 = (C - 1)(C\tilde{C}_\sigma)^{-1}$ , the first term on the right-hand side of the inequality (3.41) vanishes. To guarantee non-positivity of the second term, the condition

$$\frac{C}{C - 1}\tilde{\mu} \leq \mu - \frac{\nu}{C}, \quad \text{with } \tilde{\mu} = \frac{\tilde{C}_\sigma^2}{4\omega_1},$$

needs to be fulfilled. The above condition has the same form as condition (3.29) for the first order IMEX scheme and can thus be fulfilled by choosing  $C \geq \frac{\nu + \mu}{\mu - \tilde{\mu}}$  if  $\tilde{\mu} < \mu$ . Setting

$$C_1 = \frac{\alpha_1}{\beta_1}C, \quad C_2 = \frac{2\alpha_2}{\beta_3}C, \quad C_3 = \frac{2\alpha_3}{\beta_3}C,$$

we may rewrite (3.42) as

$$\Delta t \frac{a^2}{d} \leq \frac{1}{C} \min \left\{ \left( \frac{\alpha_1^2}{\beta_1} + \frac{2\alpha_2^2}{\beta_3} \right)^{-1}, \frac{\theta\beta_3}{2\alpha_3^2} \right\}, \quad C = \frac{\nu + \mu}{\mu - \tilde{\mu}}. \quad (3.43)$$

We summarize and obtain the following

**Theorem 3.5.** *Let  $N \in \mathbb{N}_0$  denote the polynomial degree of the DG approximation space  $V_h$  defined in (2.2). Let  $\sigma, \mu \in \mathbb{R}$  fulfill the condition*

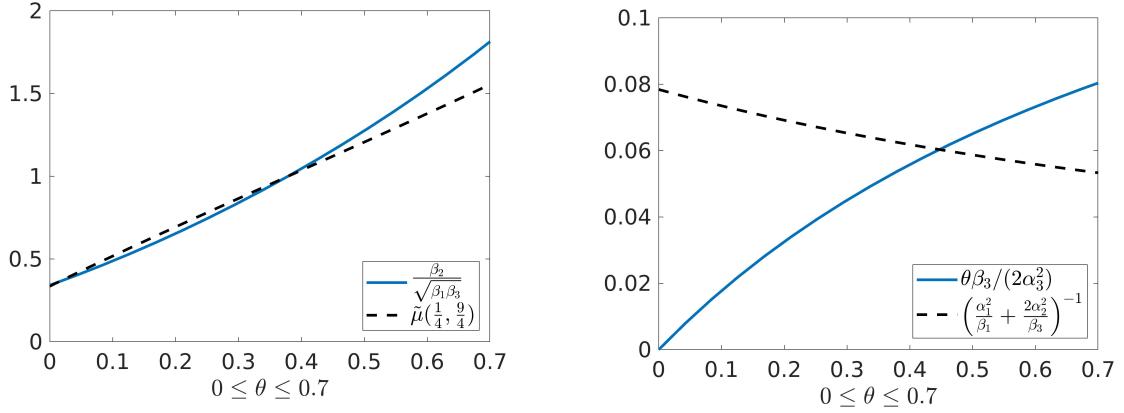
$$\tilde{\mu} := \frac{\left( |1 - \sigma| + |1 + \sigma| \frac{\beta_2}{\sqrt{\beta_1\beta_3}} \right)^2}{4\omega_1} < \mu,$$

where  $\omega_1$  denotes the first weight of the Legendre-Gauss-Lobatto quadrature with  $N + 1$  nodes and  $\beta_1, \beta_2, \beta_3$  are defined by (3.40) for  $\theta \in [0, 0.7]$  suitably chosen.

Then, the IMEX-DG scheme (3.34), (3.35) with the  $(\sigma, \mu)$ -diffusion operator  $\mathcal{L}_j$  defined in (2.15) fulfills  $\|u^{n+1}\| \leq \|u^n\|$  if the time step is bounded by the conditions (3.43) with  $\alpha_1, \alpha_2, \alpha_3$  as in (3.39).

The condition  $\tilde{\mu} < \mu$  is again a restriction on the parameters  $\sigma$  and  $\mu$  of the chosen scheme within the  $(\sigma, \mu)$ -family and thus classifies the admissible members within this family. Different from the first order IMEX scheme, admissibility is now dependent on  $\tilde{C}_\sigma$  which in turn depends on  $\beta_1, \beta_2, \beta_3$  and thus on the free parameter  $\theta \in [0, 0.7]$ . Only for  $\sigma = -1$ , yielding the sub-family of symmetric  $(\sigma, \mu)$ -schemes, we have  $\tilde{C}_\sigma = 2$ , independent of  $\theta$ . For  $\sigma \neq -1$ , monotonicity arguments show  $\tilde{C}_\sigma$  to be strictly increasing. In fact, the factor  $\beta_2/\sqrt{\beta_1\beta_3}$ , depicted in Fig. 3.4a, is strictly increasing with  $\theta$ . This also means that the reference quantity  $\tilde{\mu}$  is strictly increasing. Therefore, the smaller we chose  $\theta$ , the larger is the range of  $(\sigma, \mu)$ -schemes allowing for unconditionally stable second order IMEX time integration independent of grid refinement.

On the other hand, for values of  $\theta \leq 0.4$  the  $\theta$ -dependent factor  $\theta\beta_3/(2\alpha_3^2)$  dictates the time step restriction (3.43) as can be seen in Fig. 3.4b showing both of the  $\theta$ -dependent factors, the minimum of which is relevant for determining the allowable time step as a function of the



(a)  $\theta$ -dependent factor in  $\tilde{C}_\sigma$  and reference quantity  $\tilde{\mu} = \frac{\tilde{C}_\sigma}{4\omega_1}$  for the  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme for  $N = 1$ .

(b)  $\theta$ -dependent factors determining the time step restriction (3.43) for grid-independent  $L^2$ -stability.

Figure 3.4: Plots of  $\theta$ -dependent parameters influencing both the range of admissible  $(\sigma, \mu)$ -schemes and the time step restriction.

advection and diffusion coefficients in (3.43). Therefore, smaller values of  $\theta$  will lead to more restrictive time step constraints with respect to the advection and diffusion coefficients. Thus, for schemes far from the symmetric case  $\sigma = -1$ , unconditional stability may be possible but at the expense of smaller time steps for a convection-dominated situation.

Analogous to the first order IMEX-DG scheme, the BR1 and Baumann-Oden diffusion schemes do not fulfill the conditions on  $\sigma$  and  $\mu$  given in Theorem 3.5. Accordingly, the numerical experiments in Section 3.3.3 show that, in general, these schemes do not allow for a grid-independent time step choice. For the symmetric  $(\sigma, \mu)$ -schemes, including the BR2 scheme and the symmetric LDG variant, the conditions on  $\sigma$  and  $\mu$  are the same as in the first-order case. Analogous to the first order IMEX-DG scheme, those latter schemes therefore admit grid-independent time step choice, only based on the relation of advection and diffusion coefficients. The  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme for a polynomial degree of  $N = 1$  also fulfills the conditions given in Theorem 3.5. For this scheme, the reference quantity  $\tilde{\mu}$  is depicted in Fig. 3.4a which shows that  $\tilde{\mu} < \mu = 9/4$  for all values of  $\theta \in [0, 0.7]$ .

#### Extension to DG schemes on non-uniform grids and to multiple space dimensions

For simplicity of presentation, only uniform grids have been considered. However, the provided results also carry over to a more general class of non-uniform grids, but the corresponding analysis is more technical. For the following discussion of this extension, the cell sizes of a non-uniform grid are denoted by  $\Delta x_j = x_{j+1} - x_j$  and the mesh width is defined as  $\Delta x = \max \Delta x_j$ .

For the IMEX-LDG scheme, the assumptions in Wang et al. [205] are already relaxed to quasi-uniform partitions which assume the existence of a positive constant  $\rho$  with  $\frac{\Delta x_j}{\Delta x} \geq \rho$  for all  $j$  as  $\Delta x \rightarrow 0$ . Hence, considering Lemma 3.2, assertions (3.21) and (3.22) hold for these non-uniform grids as well.

The  $(\sigma, \mu)$  scheme on non-uniform grids may be defined by

$$\begin{aligned}\mathcal{L}_j(u, v) = & -(u_x, v_x)_j + \{u_x\}_{j+1}v_j^- - \{u_x\}_jv_j^+ + \frac{\sigma}{2} \left( (v_x[u])_{j+1} + (v_x^+[u])_j \right) \\ & + \frac{\mu}{2} \left( \left( \frac{1}{\Delta x_j} + \frac{1}{\Delta x_{j+1}} \right) ([u]v^-)_{j+1} - \left( \frac{1}{\Delta x_{j-1}} + \frac{1}{\Delta x_j} \right) ([u]v^+)_j \right),\end{aligned}$$

which correspondingly modifies the term in (2.20) containing  $\frac{\mu}{\Delta x}$  while assertion (2.21) becomes

$$\left| \sum_{j=1}^E \{u_x\}_j [v]_j \right| \leq \tilde{C} \|u_x\|^2 + \frac{1}{8\tilde{C}\omega_1} \sum_{j=1}^E \left( \frac{1}{\Delta x_{j-1}} + \frac{1}{\Delta x_j} \right) [v]_j^2.$$

Since the modifications in (2.20) and (2.21) present the main differences with respect to the  $(\sigma, \mu)$ -scheme, the theoretical stability analysis of this work may accordingly be transferred to non-uniform meshes.

Considering the multi-dimensional case, an extension of the  $(\sigma, \mu)$ -family to uniform tensor-product grids can be found in [163]. However, if we use tensor-product integration for the multi-dimensional volume term  $(\nabla u, \nabla u)$ , as in the proof of Lemma 3.2 for the one-dimensional case, the numerical integration on Legendre-Gauss-Lobatto points is not exact. Remarking that the integration rule in the proof of Lemma 3.2 only serves the analysis and is not an integral part of the scheme, a remedy may be to use Legendre-Gauss-Lobatto integration in one coordinate direction and classical Legendre-Gauss integration in the remaining ones, running through all coordinate directions in this process. This modification for the multi-dimensional case might clear the way to yield a multi-dimensional version of estimate (2.21) and allow for an extension to the practically relevant multi-dimensional case in future work. In addition, since practical applications often contain simplicial meshes, it would be beneficial to investigate how the analysis in this work can be extended to unstructured triangular grids. Presumably, a transfer of the stability results to a certain extent is possible since a corresponding analysis is given for the IMEX-LDG scheme in [208].

### 3.3.3 Numerical results

In this section, we first compare stability and accuracy of the second order IMEX-DG schemes with respect to the discretization of the diffusion terms in case of linear PDEs. Afterwards, we study the performance of different diffusion discretizations when using a second and third order IMEX-RK schemes in combination with higher order discretizations in space. Finally, as a non-linear test case, the behavior of various IMEX-DG schemes for the viscous Burgers' equation is investigated.

#### Linear advection-diffusion with exponentially decaying solution

First, we consider the exact solution

$$U(x, t) = e^{-dt} \sin(x - at)$$

to the linear advection-diffusion equation (2.35) in the interval  $(x_a, x_b) = (-\pi, \pi)$ . The advection-diffusion problem is discretized in space by the second order nodal DG scheme

$E$	LDG	Recovery	BR1	BR2	BO
20	2.42	2.42	3.23e-01	2.45	4.50e-01
40	2.41	2.41	1.57e-01	2.42	2.35e-01
80	2.41	2.41	7.91e-02	2.41	8.89e-02
160	2.41	2.41	3.03e-02	2.41	4.01e-02
320	2.41	2.41	1.08e-02	2.41	2.05e-02
640	2.41	2.41	9.26e-03	2.41	9.87e-03
$E$	$\sigma = -1$ $\mu = 10$	$\sigma = -1$ $\mu = 1.5$	$\sigma = 0.25$ $\mu = 0.5$	$\sigma = 0.25$ $\mu = 1$	$\sigma = 0.25$ $\mu = 10$
20	2.45	7.52e-01	2.34	2.39	2.45
40	2.42	2.42	2.39	2.40	2.42
80	2.41	2.41	2.40	2.41	2.41
160	2.41	2.41	2.41	2.41	2.41
320	2.41	2.41	2.41	2.41	2.41
640	2.41	2.41	2.41	2.41	2.41

Table 3.2: Values of  $\tau = \frac{a^2}{d} \Delta t_{max}$ , where  $\Delta t_{max}$  is the maximum time step to ensure a non-increasing  $L^2$ -norm for  $d = 0.1, a = 0.1$ .

on Legendre-Gauss-Lobatto nodes, hence the polynomial degree is  $N = 1$ . As in the theoretical investigation, advection terms are discretized by upwind fluxes and diffusion terms are discretized either by the LDG scheme or by various members of the  $(\sigma, \mu)$ -family. The second order IMEX scheme (3.17) is then used to discretize advection terms explicitly and diffusion terms implicitly.

From the theoretical analysis, we expect the schemes to be stable for time steps  $\Delta t \leq \frac{\tau d}{a^2}$ , where  $\tau$  is some constant independent of grid refinement.

Tables 3.2, 3.3 and 3.4 show the analysis of the maximum stable time step for different advection and diffusion parameters  $a$  and  $d$ , where we vary the number of cells  $E$  and compute the numerical solution until the final time  $T = 1000$ . The maximum stable time step  $\Delta t_{max}$  is determined as the maximum time step for which the  $L^2$ -norm of the numerical solution is non-increasing and the corresponding values of  $\tau = \frac{a^2}{d} \Delta t_{max}$  are listed. As predicted by the theoretical analysis, the BR1 and Baumann-Oden(BO) scheme do not admit grid-independent time step sizes. Here, we clearly observe the behavior  $\tau = \mathcal{O}(\Delta x)$  for the admissible time step, analogous to the time step restrictions for explicitly discretized advection equations. For the other investigated diffusion schemes the values of  $\tau$  nearly coincide for moderate sizes of the diffusion coefficient (Table 3.2). For small sizes of the diffusion coefficient (Table 3.3), on coarse grids, the allowable time step size scales with the grid size and those members of the  $(\sigma, \mu)$ -family with a large value of  $\mu$  admit much larger time steps. On fine grids, the allowable time step sizes almost coincide again, except for the BR1 and BO discretizations.

For the diffusion-dominated case, the time step restrictions with respect to the classical DG diffusion discretizations are indicated in Table 3.4. Here, the implicit discretization of the dominant diffusion term yields increased stability except for the BR1 and BO diffusion discretization. Hence, for LDG, BR2 and the Recovery discretization, no time step restriction is needed for this particular test case.

$E$	LDG	Recovery	BR1	BR2	BO
20	6.63	6.82	6.30	6.87	6.42
40	3.45	3.78	3.15	3.80	3.21
80	1.96	2.53	1.57	2.73	1.66
160	1.40	1.77	7.82e-01	2.17	9.03e-01
320	1.50	1.48	3.92e-01	1.70	5.01e-01
640	1.42	1.42	1.96e-01	1.50	2.84e-01
<hr/>					
$E$	$\sigma = -1$	$\sigma = -1$	$\sigma = 0.25$	$\sigma = 0.25$	$\sigma = 0.25$
	$\mu = 10$	$\mu = 1.5$	$\mu = 0.5$	$\mu = 1$	$\mu = 10$
20	1.05e+01	6.46	6.45	6.53	1.01e+01
40	6.22	3.30	3.24	3.35	5.95
80	3.80	1.75	1.69	1.88	3.62
160	2.39	1.01	1.01	1.32	2.28
320	1.75	6.93e-01	9.38e-01	1.24	1.69
640	1.50	1.49	1.25	1.34	1.48

Table 3.3: Values of  $\tau = \frac{a^2}{d} \Delta t_{max}$ , where  $\Delta t_{max}$  is the maximum time step to ensure a non-increasing  $L^2$ -norm for  $d = 0.01, a = 0.2$ .

$E$	LDG	Recovery	BR1	BR2	BO
20	+	+	6.65e-02	+	7.82e-02
40	+	+	3.23e-02	+	3.43e-02
80	+	+	1.57e-02	+	1.57e-02
160	+	+	6.94e-03	+	7.91e-03
320	+	+	3.03e-03	+	3.03e-03
640	+	+	1.08e-03	+	1.08e-03

Table 3.4: Values of  $\tau = \frac{a^2}{d} \Delta t_{max}$ , where  $\Delta t_{max}$  is the maximum time step to ensure a non-increasing  $L^2$ -norm for  $d = 0.5, a = 0.1$  an entry of “+” means that for this test case, the scheme is unconditionally stable independent of the time step.

Tables 3.5 and 3.6 show the  $L^2$ -errors and the experimental order of accuracy for the various schemes applied to this problem. Here, we compute the numerical solution until final time  $T = 100$  and use both moderate time steps  $\Delta t = 5\Delta x$  (first part of Table 3.5) and large time steps  $\Delta t = 25\Delta x$  (second part of Table 3.5) for the parameters  $d = 0.1$ ,  $a = 0.1$ . For the parameters  $d = 0.1$ ,  $a = 1$ , we set  $\Delta t = \Delta x$  to achieve stability. The corresponding  $L^2$ -errors and the derived experimental orders of accuracy are listed in Table 3.6. Basically in all cases, a second order convergence rate is experimentally confirmed except for the poor performance of the BR1 scheme which is unstable for the larger time step of  $\Delta t = 25\Delta x$  (Table 3.5). In addition, the BR1 scheme shows significantly larger errors for the convection-dominated case (Table 3.6).

### Linear advection-diffusion with exponentially growing solution

As in [205], we now supplement the linear advection-diffusion equation (2.35) by a source term, i.e.

$$\begin{aligned} U_t + U_x &= dU_{xx} + g(x, t), \quad (x, t) \in Q = \Omega \times (0, T), \quad \Omega = (-\pi, \pi), \\ U(x, 0) &= \sin x, \quad x \in (-\pi, \pi), \end{aligned} \tag{3.44}$$

where the source term  $g(x, t) = e^{dt} (2d \sin x + \cos x)$  is chosen such that the exact solution is  $U(x, t) = e^{dt} \sin x$  which is exponentially growing in time.

The computations are carried out until the final time  $T = 10$  with time steps  $\Delta t = \Delta x$ . The corresponding results for diffusion parameters  $d = 0.1$  and  $d = 1$  are shown in Tables 3.7 and 3.8, respectively. Again, all schemes exhibit second-order convergence behavior. Different from the previous example, the  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme is the most accurate one in this case. As mentioned before, this scheme has in fact been constructed to achieve a 4th-order truncation error. Its superior performance for this test case suggests that it may be beneficial to further consider members of the  $(\sigma, \mu)$ -family of diffusion discretizations in combination with IMEX time integration and that promising family members may be non-symmetric.

$\Delta t = 5\Delta x$									
$E$	LDG		Recovery		BR1		BR2		
	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	
20	2.38e-05		1.87e-06		8.78e-06		9.65e-06		
40	5.79e-06	2.04	4.44e-07	2.07	2.40e-06	1.88	2.45e-06	1.98	
80	1.46e-06	1.99	1.09e-07	2.03	6.23e-07	1.95	6.21e-07	1.98	
160	3.67e-07	1.99	2.69e-08	2.02	1.58e-07	1.98	1.56e-07	1.99	
320	9.24e-08	1.99	6.69e-09	2.01	3.98e-08	1.99	3.92e-08	1.99	
640	2.32e-08	1.99	1.67e-09	2.01	9.99e-09	1.99	9.82e-09	1.99	
$E$	$\sigma = -1, \mu = 10$		$\sigma = -1, \mu = 1.5$		$\sigma = 0.25, \mu = 0.5$		$\sigma = 0.25, \mu = 10$		
	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	
20	9.72e-06		9.49e-06		1.85e-05		7.40e-06		
40	2.46e-06	1.98	2.45e-06	1.95	5.12e-06	1.85	1.87e-06	1.98	
80	6.21e-07	1.99	6.20e-07	1.98	1.36e-06	1.91	4.73e-07	1.98	
160	1.56e-07	1.99	1.56e-07	1.99	3.51e-07	1.95	1.19e-07	1.99	
320	3.92e-08	1.99	3.92e-08	1.99	8.92e-08	1.98	2.99e-08	1.99	
640	9.82e-09	2.00	9.82e-09	2.00	2.25e-08	1.99	7.50e-09	2.00	
$\Delta t = 25\Delta x$									
$E$	LDG		Recovery		BR1		BR2		
	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	
20	1.25e-04		1.06e-04		-		1.00e-04		
40	3.01e-05	2.06	3.37e-05	1.65	-		3.20e-05	1.65	
80	7.74e-06	1.96	8.95e-06	1.91	-		8.55e-06	1.90	
160	1.91e-06	2.02	2.24e-06	1.99	-		2.15e-06	1.99	
320	4.72e-07	2.02	5.56e-07	2.01	-		5.34e-07	2.01	
640	1.17e-07	2.01	1.38e-07	2.01	-		1.33e-07	2.01	
$E$	$\sigma = -1, \mu = 10$		$\sigma = -1, \mu = 1.5$		$\sigma = 0.25, \mu = 0.5$		$\sigma = 0.25, \mu = 10$		
	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	EOC	
20	1.00e-04		3.36e-02		1.29e-03		1.02e-04		
40	3.20e-05	1.64	3.20e-05	10.04	3.82e-05	5.08	3.24e-05	1.65	
80	8.55e-06	1.90	8.55e-06	1.90	9.90e-06	1.95	8.65e-06	1.91	
160	2.15e-06	1.99	2.15e-06	1.99	2.47e-06	2.00	2.17e-06	2.00	
320	5.34e-07	2.01	5.34e-07	2.01	6.13e-07	2.01	5.39e-07	2.01	
640	1.33e-07	2.01	1.33e-07	2.01	1.52e-07	2.01	1.34e-07	2.01	

Table 3.5: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) with respect to the diffusion fluxes for the parameters  $d = 0.1$ ,  $a = 0.1$ . Computations carried out until final time  $T = 100$  with time steps  $\Delta t = 5\Delta x$  and  $\Delta t = 25\Delta x$ .

$E$	LDG		Recovery		BR1		BR2	
	$L^2$ -error	EOC						
20	1.11e-05		6.74e-05		5.14e-03		8.90e-06	
40	2.59e-06	2.10	1.93e-05	1.81	1.07e-03	2.27	4.31e-07	4.37
80	6.40e-07	2.02	5.42e-06	1.83	3.15e-04	1.76	2.14e-07	1.01
160	1.59e-07	2.01	1.45e-06	1.90	6.16e-05	2.35	9.90e-08	1.11
320	3.98e-08	2.00	3.76e-07	1.95	2.83e-05	1.12	3.13e-08	1.66
640	9.95e-09	2.00	9.56e-08	1.97	6.50e-06	2.12	8.71e-09	1.85

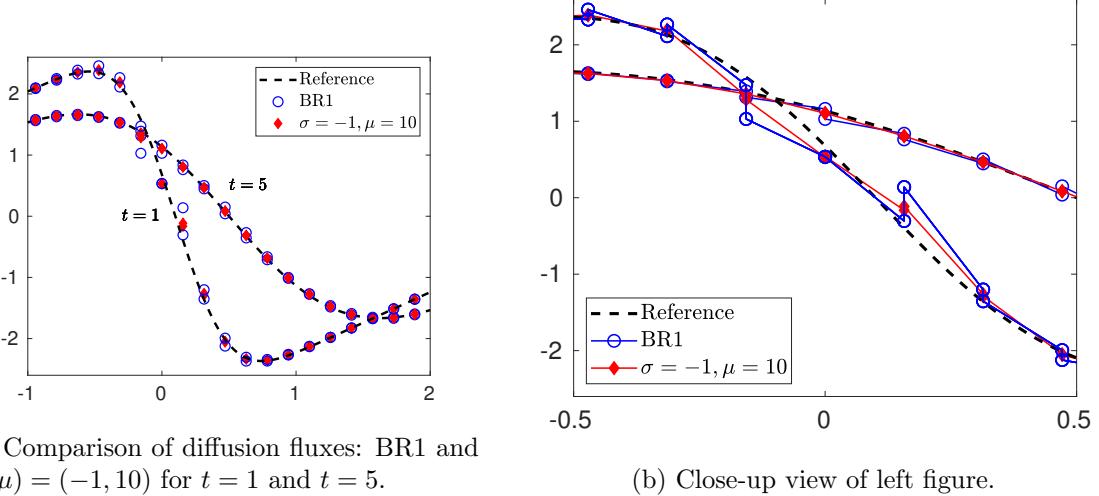
Table 3.6: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) with respect to the classical diffusion fluxes for the parameters  $d = 0.1$ ,  $a = 1$ . Computations carried out until final time  $T = 100$  with time step  $\Delta t = \Delta x$ .

$E$	LDG		Recovery		BR1		BR2	
	$L^2$ -error	EOC						
20	1.48e-01		6.65e-02		1.62e-01		1.03e-01	
40	3.70e-02	2.00	1.20e-02	2.48	4.01e-02	2.02	2.36e-02	2.13
80	9.27e-03	1.99	2.23e-03	2.42	1.01e-02	1.99	5.63e-03	2.07
160	2.32e-03	1.99	4.56e-04	2.29	2.53e-03	2.00	1.38e-03	2.02
320	5.81e-04	1.99	1.02e-04	2.17	6.37e-04	1.99	3.44e-04	2.01
640	1.45e-04	1.99	2.40e-05	2.08	1.57e-04	2.01	8.60e-05	2.00

Table 3.7: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) with respect to the classical diffusion fluxes for the exponentially growing solution of (3.44) with  $d = 0.1$ . Computations carried out until final time  $T = 10$  with time step  $\Delta t = \Delta x$ .

$E$	LDG		Recovery		BR1		BR2	
	$L^2$ -error	EOC						
20	1.12e+03		3.69e+02		9.06e+02		5.63e+02	
40	2.95e+02	2.00	9.84e+01	2.48	2.55e+02	2.02	1.50e+02	2.13
80	7.55e+01	1.99	2.65e+01	2.42	6.84e+01	1.99	3.97e+01	2.07
160	1.91e+01	1.99	6.81e+00	2.29	1.77e+01	2.00	1.01e+01	2.02
320	4.80e+00	1.99	1.74e+00	2.17	4.50e+00	1.99	2.57e+00	2.01
640	1.20e+00	1.99	4.37e-01	2.08	1.13e+00	2.02	6.47e-01	2.00

Table 3.8: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) with respect to the classical diffusion fluxes for the exponentially growing solution of (3.44) with  $d = 1$ . Computations carried out until final time  $T = 10$  with time step  $\Delta t = \Delta x$ .



(a) Comparison of diffusion fluxes: BR1 and  $(\sigma, \mu) = (-1, 10)$  for  $t = 1$  and  $t = 5$ .

(b) Close-up view of left figure.

Figure 3.5:  $\text{DG}(N = 1)$  solution for discontinuous initial condition (3.45).

### Behavior for discontinuous solutions

Next, in order to investigate the behavior of the second order DG schemes for discontinuous initial conditions, we consider the initial condition

$$U(x, 0) = \begin{cases} x + \pi & \text{for } x > 0, \\ x - \pi & \text{for } x < 0, \end{cases} \quad (3.45)$$

on the interval  $(x_a, x_b) = (-\pi, \pi)$ . This initial condition then evolves according to the linear advection-diffusion equation (2.35).

Figure 3.5 depicts the output of the  $\text{DG}(N = 1)$  schemes on  $E = 40$  elements using the BR1 diffusion discretization with  $\sigma = -1$  and  $\mu = 1$  as well as the  $(\sigma, \mu)$  diffusion discretization with  $\sigma = -1$  and  $\mu = 10$ . The second order IMEX scheme (3.17) is used for time integration with time step  $\Delta t = 10\Delta x$ . A reference solution is computed using the fourth order  $\text{DG}(N = 3)$  scheme with LDG diffusion discretization on  $E = 640$  cells. From Figure 3.5b, we may observe that in the smooth profile which develops out of the initial jump, the BR1 discretization yields larger errors than the more stable variant with  $\mu = 10$  and seems to oscillate around the profile. However, both variants of the DG scheme capture the reference solution to this advection-diffusion problem with sufficient accuracy as shown in Figure 3.5a.

### Performance for higher order schemes

The application of a higher order discretization in space when choosing a polynomial degree of  $N > 1$  is usually combined with a higher order time discretization. In this work, the theoretical stability analysis was carried out for particular IMEX time integrators of first and second order. Using these time discretization schemes leads to only first or second order of the fully discrete scheme.

Nevertheless, numerical experiments for the cases  $N = 2, 3$  will be reported in this section, where the DG space discretization is complemented by the second order IMEX scheme (3.17) as well as the third order IMEX scheme taken from [30] which is given by:

$$\begin{array}{c|ccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \gamma & \gamma & 0 & 0 & 0 & 0 & \gamma & 0 & 0 \\ \frac{1+\gamma}{2} & \frac{1+\gamma}{2} - a_1 & a_1 & 0 & 0 & 0 & \frac{1-\gamma}{2} & \gamma & 0 \\ 1 & 0 & 1 - a_2 & a_2 & 0 & 0 & b_1 & b_2 & \gamma \\ \hline & 0 & b_1 & b_2 & \gamma & 0 & b_1 & b_2 & \gamma \end{array} \quad (3.46)$$

with

$$\begin{aligned} \gamma &\approx 0.435866521508459 \text{ the middle root of } 6x^3 - 18x^2 + 9x - 1, \\ b_1 &= -1.5\gamma^2 + 4\gamma - 0.25, \\ b_2 &= 1.5\gamma^2 - 5\gamma + 1.25, \\ a_1 &= -0.35, \\ a_2 &= \frac{1/3 - 2\gamma^2 - 2b_2 a_1 \gamma}{\gamma(1-\gamma)}. \end{aligned}$$

Tables 3.9, 3.10, 3.11 and 3.12 show the numerical results for  $N = 2, 3$  for the first test case in this section, solving the advection-diffusion equation (2.35) in the interval  $(x_a, x_b) = (-\pi, \pi)$  with periodic boundary conditions and initial condition  $U(x, 0) = \sin(x)$ . Hereby, Tables 3.9 and 3.10 provide a numerical analysis of the allowable time step size for IMEX time integration. Again, the BR1 diffusion discretization does not admit a grid independent time step choice as predicted by the theoretical analysis. However, for the Baumann-Oden method, the time step restriction is comparable to those members of the  $(\sigma, \mu)$ -family which respect the condition given in Theorem 3.5. The observed grid-independent stability of the second order IMEX-DG scheme with Baumann-Oden diffusion discretization for this particular test case is unexpected but does not contradict the theory which only provides upper bounds. Tables 3.11 and 3.12 provide the comparison of the higher order schemes with respect to accuracy. It is shown that due to its less favorable stability properties, the BR1 diffusion discretization yields large  $L^2$ -errors on fine grids. For the other diffusion discretizations, the expected order of accuracy is achieved. Obviously, the order of accuracy of the fully discrete scheme is restricted to the order of accuracy of the IMEX time integrator.

$N$	$E$	LDG	$\sigma = \frac{1}{4}, \mu = \frac{9}{4\omega_1}$	BR1	BR2	BO
$d = 0.1, a = 0.1$						
2	10	2.41	2.42	2.63e-01	2.42	2.45
	20	2.41	2.42	1.35e-01	2.41	2.43
	40	2.41	2.41	7.09e-02	2.41	2.42
	80	2.41	2.41	3.67e-02	2.41	2.41
	160	2.41	2.41	1.84e-02	2.41	2.41
	320	2.41	2.41	9.26e-03	2.41	2.41
$d = 0.01, a = 0.2$						
2	10	5.28	6.13	4.75	5.96	5.00
	20	2.93	3.98	2.36	3.89	2.62
	40	1.88	2.48	1.18	2.80	1.50
	80	1.72	1.72	5.99e-01	1.85	9.39e-01
	160	1.44	1.49	3.08e-01	1.49	1.20
	320	1.41	1.44	1.61e-01	1.42	1.53
3	10	3.38	5.04	2.66	5.27	3.02
	20	2.22	3.03	1.34	3.27	1.72
	40	1.85	2.10	6.86e-01	2.32	1.25
	80	1.45	1.92	3.60e-01	1.87	2.19
	160	1.41	1.81	1.96e-01	1.79	1.87
	320	1.41	1.78	1.09e-01	1.78	1.81

Table 3.9: Stability analysis for the DG scheme with  $N = 2, 3$  and second order IMEX time integration (3.17): Values of  $\tau = \frac{a^2}{d} \Delta t_{max}$ , where  $\Delta t_{max}$  is the maximum time step to ensure a non-increasing  $L^2$ -norm.

$N$	$E$	LDG	$\sigma = \frac{1}{4}, \mu = \frac{9}{4\omega_1}$	BR1	BR2	BO
$d = 0.1, a = 0.1$						
2	10	5.86	5.88	2.72e-01	5.87	5.97
	20	5.86	5.87	1.36e-01	5.86	5.88
	40	5.86	5.86	6.91e-02	5.86	5.66
	80	5.86	5.86	3.49e-02	5.86	5.52
	160	5.86	5.86	1.72e-02	5.86	5.68
	320	5.86	5.86	8.64e-03	5.86	5.80
3	10	5.86	5.86	1.57e-01	5.86	5.86
	20	5.86	5.86	8.07e-02	5.86	5.86
	40	5.86	5.86	4.10e-02	5.86	5.86
	80	5.86	5.86	2.02e-02	5.86	5.86
	160	5.86	5.86	1.05e-02	5.86	5.86
	320	5.86	5.86	4.98e-03	5.86	5.86
$d = 0.01, a = 0.2$						
2	10	6.48	7.86	6.00	7.90	6.32
	20	3.52	5.65	2.78	5.81	3.44
	40	2.20	8.38	1.34	7.91	2.41
	80	2.28	7.25	6.62e-01	7.01	4.46
	160	6.86	6.91	3.32e-01	6.84	6.57
	320	6.84	6.73	1.68e-01	6.75	6.94
3	10	4.26	7.82	2.91	7.32	3.69
	20	2.51	7.10	1.45	6.82	2.63
	40	2.04	6.91	7.29e-01	6.50	4.27
	80	4.66	6.75	3.71e-01	6.73	6.95
	160	6.62	6.65	1.91e-01	6.69	6.79
	320	6.80	6.58	9.87e-02	6.60	6.63

Table 3.10: Stability analysis for the DG scheme with  $N = 2, 3$  and third order IMEX time integration (3.46): Values of  $\tau = \frac{a^2}{d} \Delta t_{max}$ , where  $\Delta t_{max}$  is the maximum time step to ensure a non-increasing  $L^2$ -norm.

$N$	$E$	LDG		$\sigma = \frac{1}{4}, \mu = \frac{9}{4\omega_1}$		BR1	BR2	
		$L^2$ -error	EOC	$L^2$ -error	EOC	$L^2$ -error	$L^2$ -error	EOC
Second order time integration IMEX2								
2	10	2.28e-05		2.55e-05		2.27e-05	2.29e-05	
	20	5.96e-06	1.94	6.85e-06	1.90	5.99e-06	5.98e-06	1.94
	40	1.50e-06	1.99	1.74e-06	1.98	4.46e-05	1.50e-06	2.00
	80	3.73e-07	2.01	4.35e-07	2.00	2.58e-02	3.73e-07	2.01
	160	9.30e-08	2.00	1.09e-07	2.00	-	9.30e-08	2.00
	320	2.32e-08	2.00	2.71e-08	2.01	-	2.32e-08	2.00
3	10	2.32e-05		2.32e-05		-	2.32e-05	
	20	5.99e-06	1.95	6.00e-06	1.95	-	5.99e-06	1.95
	40	1.50e-06	2.00	1.50e-06	2.00	-	1.50e-06	2.00
	80	3.73e-07	2.01	3.73e-07	2.01	-	3.73e-07	2.01
	160	9.30e-08	2.00	9.30e-08	2.00	-	9.30e-08	2.00
	320	2.32e-08	2.00	2.32e-08	2.00	-	2.32e-08	2.00
Third order time integration IMEX3								
2	10	2.48e-06		6.38e-06		2.46e-06	2.61e-06	
	20	3.75e-07	2.73	1.44e-06	2.15	4.29e-01	3.84e-07	2.76
	40	5.16e-08	2.86	3.30e-07	2.13	-	5.21e-08	2.88
	80	6.74e-09	2.94	7.84e-08	2.07	-	6.75e-09	2.95
	160	8.62e-10	2.97	1.90e-08	2.04	-	8.61e-10	2.97
	320	1.09e-10	2.98	4.69e-09	2.02	-	1.09e-10	2.98
3	10	2.93e-06		2.94e-06		-	2.93e-06	
	20	4.05e-07	2.85	4.05e-07	2.86	-	4.05e-07	2.85
	40	5.33e-08	2.93	5.34e-08	2.92	-	5.33e-08	2.93
	80	6.83e-09	2.96	6.84e-09	2.96	-	6.83e-09	2.96
	160	8.66e-10	2.98	8.66e-10	2.98	-	8.66e-10	2.98
	320	1.09e-10	2.99	1.09e-10	2.99	-	1.09e-10	2.99

Table 3.11: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) for polynomial degrees of  $N = 2, 3$  and advection-diffusion parameters  $d = 0.1$ ,  $a = 0.1$ . Computations carried out until final time  $T = 100$  with time step  $\Delta t = 5\Delta x$ . An entry of “-” means that for this test case the  $L^2$ -error exceeds 1.

$N$	$E$	LDG		$\sigma = \frac{1}{4}, \mu = \frac{9}{4\omega_1}$		BR1 $L^2$ -error	BR2	
		$L^2$ -error	EOC	$L^2$ -error	EOC		$L^2$ -error	EOC
Second order time integration IMEX2								
2	10	2.25e-02		1.31e-02		-	1.32e-02	
	20	2.85e-03	2.98	1.31e-03	3.32	-	1.05e-03	3.65
	40	3.63e-04	2.97	2.20e-04	2.57	-	1.10e-04	3.25
	80	4.80e-05	2.92	5.16e-05	2.09	-	1.98e-05	2.47
	160	7.14e-06	2.75	1.29e-05	2.00	-	4.58e-06	2.11
	320	1.32e-06	2.44	3.24e-06	1.99	-	1.12e-06	2.03
3	10	8.56e-04		5.83e-04		-	6.03e-04	
	20	1.13e-04	2.92	1.08e-04	2.43	-	1.04e-04	2.54
	40	2.58e-05	2.13	2.60e-05	2.05	-	2.57e-05	2.02
	80	6.43e-06	2.00	6.45e-06	2.01	-	6.42e-06	2.00
	160	1.61e-06	2.00	1.61e-06	2.00	-	1.61e-06	2.00
	320	4.02e-07	2.00	4.02e-07	2.00	-	4.02e-07	2.00
Third order time integration IMEX3								
2	10	2.24e-02		1.31e-02		1.18e-02	1.30e-02	
	20	2.82e-03	2.99	1.28e-03	3.36	-	9.82e-04	3.73
	40	3.54e-04	2.99	2.06e-04	2.64	-	7.80e-05	3.65
	80	4.44e-05	3.00	4.78e-05	2.11	-	7.64e-06	3.35
	160	5.55e-06	3.00	1.19e-05	2.01	-	8.88e-07	3.10
	320	6.94e-07	3.00	2.99e-06	1.99	-	1.09e-07	3.03
3	10	7.52e-04		3.38e-04		-	4.45e-04	
	20	4.77e-05	3.98	1.53e-05	4.47	-	1.88e-05	4.57
	40	3.05e-06	3.97	9.91e-07	3.95	-	9.14e-07	4.36
	80	2.04e-07	3.90	9.88e-08	3.33	-	8.68e-08	3.40
	160	1.57e-08	3.70	1.14e-08	3.12	-	1.06e-08	3.03
	320	1.51e-09	3.38	1.37e-09	3.06	-	1.32e-09	3.01

Table 3.12: Comparison of  $L^2$ -errors and experimental order of convergence (EOC) for the exponentially growing solution of (3.44) with  $d = 0.1$ . Polynomial degrees of  $N = 2, 3$ . Computations carried out until final time  $T = 10$  with time step  $\Delta t = 0.5\Delta x$  for  $N = 2$  and  $\Delta t = 0.3\Delta x$  for  $N = 3$ . An entry of “-” means that for this test case the  $L^2$ -error exceeds 1.

### Numerical results for the viscous Burgers' equation

Finally, we study the behavior of IMEX-DG schemes with respect to different DG diffusion discretizations solving the viscous Burgers' equation

$$U_t(x, t) + \left( \frac{1}{2} U^2(x, t) \right)_x = d U_{xx}(x, t), \quad d = 0.1,$$

supplemented by the initial condition

$$U(x, 0) = U_0(x) = \sin x, \quad x \in [-\pi, \pi]$$

and periodic boundary conditions.

Figure 3.6 shows the initial solution as well as the numerical solution at time  $t = 2$  using the DG( $N = 2$ ) space discretization with BR2( $\eta_e = 3$ ) diffusion fluxes and applying the second order IMEX time integration method (3.17) with a time step size of  $\Delta t = 0.2$ . The same time step is used both on coarse and fine grids with  $K = 10, 50, 200, 1000$  elements, respectively. Clearly, refining the grid does not necessitate choosing smaller time steps for this combination of IMEX time integration and DG diffusion fluxes. Replacing BR2 fluxes by the LDG diffusion discretization, the same favorable behavior is shown in Figure 3.7.

On the other hand, replacing the diffusion discretization by the BR1 fluxes, a time step of  $\Delta t = 0.2$  already leads to instability for  $K = 50$  as shown in Figure 3.8. In fact, on fine grids, the simulation needs to be stopped before reaching the final time of  $t = 2$  due to the lack of stability. Even reducing the time step by a factor of 10 to  $\Delta t = 0.02$ , the results in Figure 3.9 for fine grids with  $K = 200$  and  $K = 1000$  demonstrate that combining the BR1 diffusion fluxes with IMEX advection-diffusion splitting is not favorable in terms of stability.

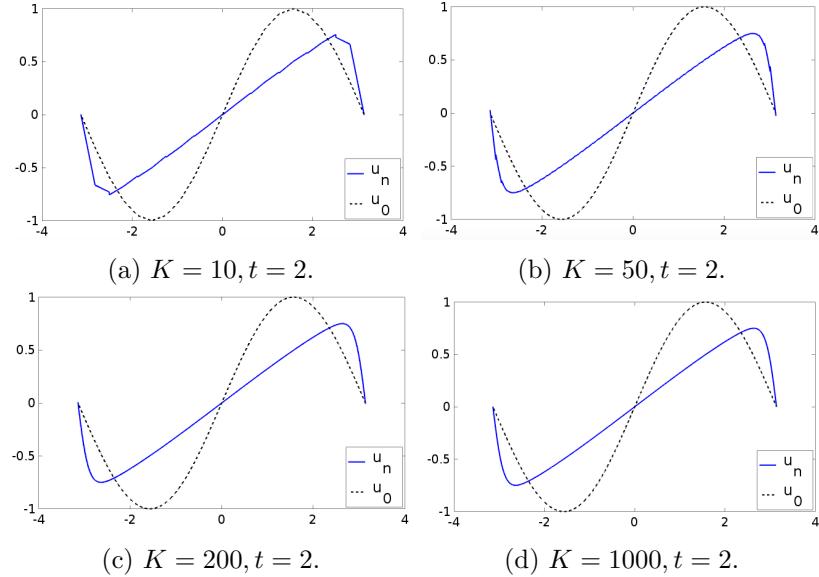


Figure 3.6: DG( $N = 2$ ) approximation of the viscous Burgers' equation using BR2( $\eta = 3$ ) diffusion fluxes and second order IMEX time integration on successively refined grids.

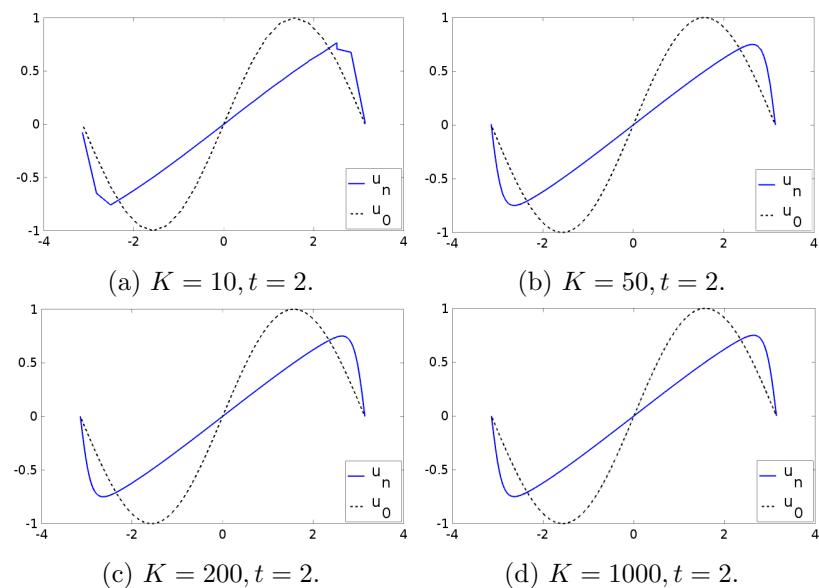


Figure 3.7: DG( $N = 2$ ) approximation of the viscous Burgers' equation using LDG diffusion fluxes and second order IMEX time integration on successively refined grids.

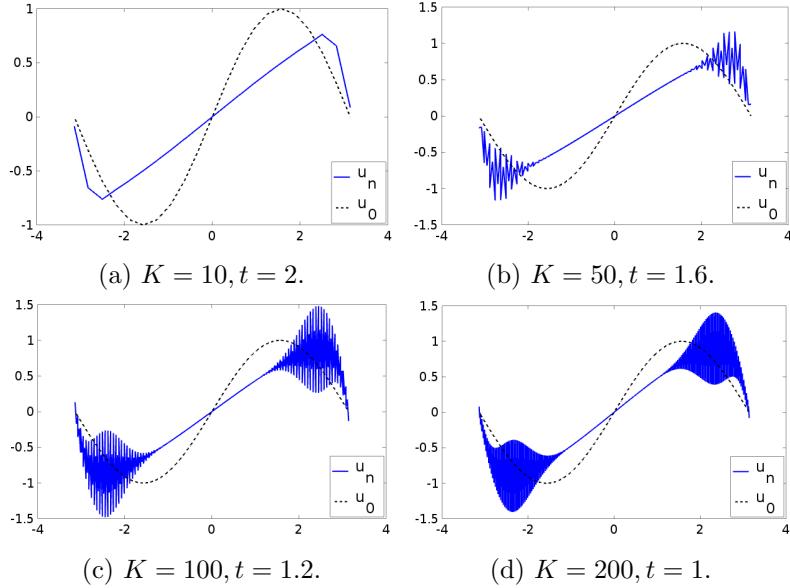


Figure 3.8:  $DG(N = 2)$  approximation of the viscous Burgers' equation using BR1 diffusion fluxes and second order IMEX time integration on successively refined grids.

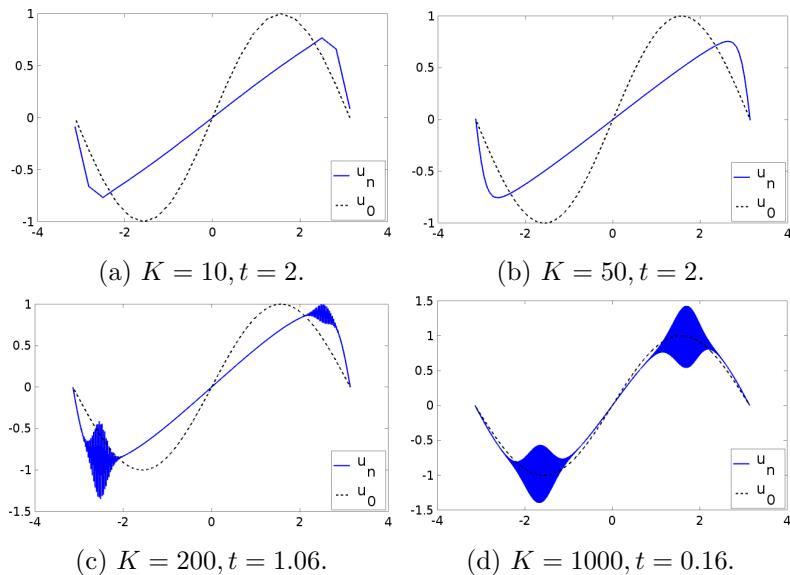


Figure 3.9:  $DG(N = 2)$  approximation of the viscous Burgers' equation using BR1 diffusion fluxes and second order IMEX time integration using small time steps of  $\Delta t = 0.02$ .

### 3.4 Positivity preservation

The quantities involved in the description of physical processes are often restricted by minimum or maximum values. For instance, the density and pressure in the description of compressible fluid flow should both be positive quantities. Regarding the flow of water in rivers, lakes or oceans, the water column height should be non-negative. This constraint also holds for the concentrations of substances in chemical processes. Furthermore, the values of mass fractions or probabilities only make sense if contained in the interval  $[0, 1]$ . In general, the approximations given by a numerical method do not necessarily satisfy these bounds. However, a violation may cause a blow-up of the numerical solution as the involved mathematical operations may not be well-defined anymore if the relevant quantities are located outside of the physically meaningful range.

In the following, we focus on the preservation of non-negativity, or short positivity, and consider initial value problems of the form

$$\mathbf{u}'(t) = \mathbf{g}(t, \mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0, \quad (3.47)$$

with the property that

$$\mathbf{u}(t) \geq 0 \text{ for all } t \geq 0 \text{ if } \mathbf{u}_0 \geq 0. \quad (3.48)$$

The solution to the above initial value problem (3.47) obviously remains positive for all  $t \geq 0$ , i.e. the positivity property (3.48) holds, if the following sufficient condition for each component of the function  $\mathbf{g}$  is fulfilled,

$$\text{if } u_i(t^*) = 0 \text{ for fixed } t^* \geq 0 \text{ then } g_i(t^*, \tilde{\mathbf{u}}) \geq 0 \text{ for all } \tilde{\mathbf{u}} \geq 0 \text{ with } \tilde{u}_i = u_i(t^*). \quad (3.49)$$

Initial value problems having the property (3.48) may also stem from the space discretization of certain partial differential equations.

In order to prevent numerical instabilities and to maintain physically meaningful approximations, we desire a numerical scheme which is *positive* or *positivity preserving*, i.e. which satisfies (3.48) in a discrete sense. The simplest approach to ensure that the numerical scheme is positive obviously consists in setting negative values to zero. However, this approach is not conservative and thus of limited applicability in the context of conservation laws where certain linear invariants need to be preserved.

Obviously, if the condition (3.49) is fulfilled then positivity is preserved by the explicit Euler scheme if time step size is sufficiently small. In fact, if  $\mathbf{u}^n \geq 0$  and the current time step  $\Delta t^n$  is restricted by

$$\Delta t^n = \min_{\{i \mid g_i(t^n, \mathbf{u}^n) < 0\}} \left| \frac{u_i}{g_i(t^n, \mathbf{u}^n)} \right|,$$

with  $\Delta t^n > 0$  due to (3.49), then the execution of an explicit Euler step yields

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t^n \mathbf{g}(t^n, \mathbf{u}^n) \geq 0.$$

For initial value problems (3.47) satisfying the condition (3.49), the positivity property of the explicit Euler scheme can be extended to the class of so-called *strong stability preserving (SSP)*

schemes reviewed in [72], thus ensuring positivity of higher order time integration schemes. Moreover, SSP schemes preserve quite general stability properties of the explicit Euler scheme defined by means of convex functions. The application of an SSP scheme is thereby based on the assumption, that there is a sufficiently small time step  $\Delta t_{FE} > 0$  such that the forward Euler scheme satisfies

$$\|\mathbf{u}^n + \Delta t \mathbf{g}(\mathbf{u}^n)\| \leq \|\mathbf{u}^n\|, \quad \forall \Delta t \leq \Delta t_{FE}. \quad (3.50)$$

where  $\|\cdot\| : V \subset \mathbb{R}^m \rightarrow \mathbb{R}$  is a given convex function. In this context, the assumption (3.50) is called the *forward Euler (FE) assumption*. Now, an SSP scheme is defined as follows.

**Definition 3.6.** A numerical method approximately solving the initial value problem (3.47) is called a strong stability preserving (SSP) method, if there is a constant  $c > 0$  such that

$$\|\mathbf{u}^{n+1}\| \leq \|\mathbf{u}^n\|, \quad \forall \Delta t \leq c \Delta t_{FE},$$

if the FE assumption (3.50) is fulfilled.

Originally, at the beginning of the development of SSP schemes, the TV semi-norm was considered as the given convex function in order to design essentially non-oscillatory methods for conservation laws. At that time, the respective time integration methods were termed TVD time integration schemes, see [178, 176]. Furthermore, since

$$\|\mathbf{u}\| = \max_i \{-u_i\}$$

obviously is a convex function, the SSP theory is also applicable to positivity preservation. Therefore, if the forward Euler method is positive with

$$\mathbf{u}^n \geq \mathbf{0} \Rightarrow \mathbf{u}^{n+1} \geq \mathbf{0} \quad \text{for } \Delta t \leq \Delta t_{FE},$$

then this property also carries over to higher order SSP schemes under the time step restriction  $\Delta t \leq c \Delta t_{FE}$ .

The first schemes having the above SSP property were members of a particular class of Runge-Kutta schemes which may be written as convex combinations of explicit Euler steps. The precise formulation of these SSP-RK schemes is

$$\begin{aligned} \mathbf{u}^{(1)} &= \mathbf{u}^n, \\ \mathbf{u}^{(i)} &= \sum_{j=1}^{i-1} \left( \alpha_{ij} \mathbf{u}^{(j)} + \Delta t \beta_{ij} \mathbf{g} \left( t^n + c_j \Delta t, \mathbf{u}^{(j)} \right) \right), \quad i = 1, \dots, s, \\ \mathbf{u}^{n+1} &= \mathbf{u}^s, \end{aligned} \quad (3.51)$$

with coefficients  $\alpha_{ij}, \beta_{ij} \geq 0$ , where  $\alpha_{ij}$  is zero only if the corresponding coefficient  $\beta_{ij}$  is zero.

For SSP-RK schemes (3.51), the requested constant  $c > 0$  in Definition 3.6 is determined by the coefficients  $\alpha_{ij}, \beta_{ij}$  occurring in the convex combinations of explicit Euler time steps determining the stage values  $\mathbf{u}^{(i)}$ . Making use of the FE assumption, we have

$$c = \min_{i,j} \frac{\alpha_{ij}}{\beta_{ij}}.$$

The SSP property of particular RK schemes also provides the mathematical background for positivity preserving DG schemes combined with high order explicit time integration for applications in gas dynamics by Zhang and Shu [219] guaranteeing positive density and pressure and in the computation of shallow water flows by Xing et al. [214] ensuring non-negative water height.

Although implicit schemes may usually take larger time steps regarding stability conditions, positivity preservation demands an additional time step constraint which often outplays the corresponding restriction due to stability. In fact, although the implicit Euler scheme is unconditionally SSP independent of the time step size as proven by Higueras [78], and hence unconditionally positive, Gottlieb et al. [72] proved that there is no Runge-Kutta scheme or linear multi-step scheme of order  $\geq 2$  which is unconditionally SSP. Therefore, if the order of accuracy of the time integration method exceeds first order, the allowable time step size guaranteeing boundedness with respect to the given convex function is still restricted by the finite SSP parameter  $c$  and the allowable time step  $\Delta t_{FE}$  of the forward Euler scheme. In addition, regarding positive schemes, a classical result by Bolley and Crouzeix [24] states that any general linear method that is unconditionally positivity preserving for all initial value problems (3.47) satisfying the positivity property (3.48) can only be first order accurate at best.

On this account, time step restrictions for positivity will be expected for higher order implicit schemes as well. However, an additional restriction of the time step size for implicit schemes to enforce positivity may lead to an inefficient scheme at the end as a lot of computational time is required to solve potentially large systems of nonlinear equations for each time step.

As a remedy for ordinary differential equations of production-destruction type, Runge-Kutta schemes may be modified by the so-called Patankar trick in order to alleviate the time step restriction based on positivity. While originally applied to explicit RK schemes, the Patankar idea has been incorporated into an SDIRK scheme in [132] to obtain an unconditionally positivity preserving scheme for the shallow water equations to be discussed in more detail in Section 4.4. Different approaches avoiding excessively small time steps for positivity are diagonally split Runge-Kutta (DSRK) methods as investigated by Horváth [80] in the context of positivity preservation or adaptive Runge-Kutta methods adapting the weights after all stage derivatives have been computed, see [145]. Similar to Patankar approaches, the DSRK methods may be unconditionally positive while avoiding a restriction to first order since they do not belong to the class of general linear methods. However, for a variety of numerical test cases, Macdonald et al. [121] demonstrated that DSRK methods may suffer from order reduction at large time step sizes behaving like first-order implicit schemes. In addition DSRK methods do not possess a built-in mechanism preserving linear invariants which is relevant for instance in order to guarantee specific conservation properties for ODE systems resulting from semi-discrete conservation laws. On the other hand, the adaptive RK methods developed in [145] based on an adaption of the RK weights after all stage derivatives have been calculated is specifically designed to preserve all linear invariants of the given ODE. These methods may also reject or reduce a given time step if no feasible combination of RK weights satisfying the positivity constraint is found. However, for many physically relevant conservation laws, the right-hand side of the system of ODEs resulting from a suitable space discretization will not be computable for arguments violating the positivity constraints, for instance due to the

evaluation of the sound speed within numerical flux functions for the Euler equations or the shallow water equations considered in Section 1.4 and Section 1.5, respectively.

In this section, we focus on the Patankar approach modifying Runge-Kutta schemes to achieve both unconditional positivity and conservation for production-destruction equations. First, we will review modified Patankar-Runge-Kutta schemes for ordinary differential equations in production-destruction form in Section 3.4.1. Furthermore, in Section 3.4.2, the behavior of specific Patankar-modified explicit Runge-Kutta schemes is investigated for semi-discretizations of classical linear partial differential equations which can be written in production-destruction form. Finally, Section 3.4.3 deals with positivity preservation in the context of implicit schemes, where we review the construction of the unconditionally positive implicit MPSDIRK3 scheme developed by Meister and Ortob in [132].

### 3.4.1 The Patankar approach applied to production-destruction equations

In the context of geobiochemical models, so-called production-destruction equations are frequently encountered. These models describe the time-evolution of non-negative quantities and often take into account some type of mass conservation. The underlying ODE systems describing the time-evolution of non-negative quantities  $\mathbf{u}(t)$  can usually be written in the form

$$\frac{du_i}{dt} = P_i(\mathbf{u}) - D_i(\mathbf{u}), \quad j = 1, \dots, I, \quad t \in \mathbb{R}^+, \quad (3.52)$$

where  $\mathbf{u} = (u_1, \dots, u_I)^T$ ,  $I \in \mathbb{N}$ , denotes the vector of non-negative constituents and  $t$  denotes the time. Furthermore, the terms  $P_i(\mathbf{u})$  and  $D_i(\mathbf{u})$  represent the production and destruction rates of the  $i$ -th constituent, respectively. Initial conditions for the system (3.52) are given by  $\mathbf{u}(0) = \mathbf{u}_0 \geq \mathbf{0}$ . The production-destruction terms may now be written as

$$P_i(\mathbf{u}) = \sum_{j=1}^I p_{ij}(\mathbf{u}), \quad D_i(\mathbf{u}) = \sum_{j=1}^I d_{ij}(\mathbf{u}), \quad (3.53)$$

where  $d_{ij}(\mathbf{u}) \geq 0$  is the rate at which the  $i$ -th constituent transforms into the  $j$ -th component, while  $p_{ij}(\mathbf{u}) \geq 0$  is the rate at which the  $j$ -th constituent transforms into the  $i$ -th component. Hence, the definition of a production-destruction equation contains the condition  $p_{ij}(\mathbf{u}) = d_{ji}(\mathbf{u})$  for  $i \neq j$ . For fully conservative production-destruction equations, we also have  $p_{ii}(\mathbf{u}) = d_{ii}(\mathbf{u}) = 0$  by definition. As also stated in [29], for non-negative initial conditions  $u_i(0) \geq 0$ ,  $i = 1, \dots, I$ , one can easily show by a simple contradiction argument that the condition

$$d_{ij}(\mathbf{u}) \rightarrow 0 \quad \text{for } u_i \rightarrow 0, \quad (3.54)$$

for all  $i, j \in \{1, \dots, I\}$ , guarantees  $\mathbf{u}(t) \geq \mathbf{0}$  for all  $t \in \mathbb{R}_0^+$ . In fact, if we assume for a smooth solution  $\mathbf{u}(t)$  of (3.52) fulfilling the initial condition  $\mathbf{u}(0) \geq \mathbf{0}$  that at least one component becomes negative, e.g.  $u_j(t^*) < 0$  at a given time  $t^*$ , then there needs to exist  $0 \leq \bar{t} \leq t^*$  with  $u_j(\bar{t}) = 0$  and  $\frac{d}{dt}u_j(\bar{t}) < 0$  contradicting the condition (3.54).

In order to obtain a well-posed problem, we furthermore require Lipschitz continuity of the right-hand side of (3.52). Hence, we have  $|d_{ij}(\mathbf{u}) - d_{ij}(\tilde{\mathbf{u}})| \leq L \cdot \|\mathbf{u} - \tilde{\mathbf{u}}\|$ . Together with

condition (3.54) this obviously yields boundedness of the quantities  $\frac{d_{ij}(\mathbf{u})}{u_i}$ . In addition, in order to be able to define the Patankar approach in the following, we actually need a slightly stronger assumption on the destruction rates. More precisely, we assume convergence of the fractions  $\frac{d_{ij}(\mathbf{u})}{u_i}$  for  $u_i \rightarrow 0$ , such that these are well-defined for  $u_i = 0$ . In the following, we will therefore assume this stronger condition to be fulfilled.

In the context of such production-destruction processes, numerical methods that compute approximations  $\mathbf{u}^n$  to the values  $\mathbf{u}(t^n)$  at time  $t^n \in \mathbb{R}_0^+$  are of course supposed to be both conservative and positive. For one-step schemes, these two properties are defined as follows.

**Definition 3.7.** *A numerical method  $\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t^n \phi(\mathbf{u}^n, t^n, \Delta t^n)$  with incremental function  $\phi$  and time step  $\Delta t^n = t^{n+1} - t^n$  applied to a given conservative production-destruction equation of the form (3.52) is called*

- conservative, if  $\sum_{i=1}^I (u_i^{n+1} - u_i^n) = 0$ ,
- unconditional positive, if  $\mathbf{u}^{n+1} \geq \mathbf{0}$  for any given previous value  $\mathbf{u}^n \geq \mathbf{0}$  and using an arbitrarily large time step  $\Delta t^n \geq 0$ .

**Remark 3.8.** *We note that this definition of unconditional positivity differs from the one used in [29] based on strict inequality, but coincides with the definition used in [80] since in Definition 3.7, the components are allowed to become zero. The methods thus become applicable for a larger class of problems, e.g. shallow water equations with vanishing water height to be discussed in Chapter 4.*

In [29], Burchard, Deleersnijder and Meister pioneered the construction of so-called modified Patankar-Runge-Kutta schemes that respect both the requirement of non-negativity and of conservation. These schemes are based on well-known Runge-Kutta methods which are suitably modified in order to fulfill both properties, resulting in non-linear schemes even if applied to linear ODEs. Originally, the Patankar trick was introduced in [155] as a source term linearization dealing with the numerical simulation of turbulent flows and guaranteeing unconditional positivity in that specific context.

Let us first note that the forward Euler scheme applied to (3.52), i.e.

$$u_i^{n+1} = u_i^n + \Delta t \left( \sum_{j=1}^I p_{ij}(\mathbf{u}^n) - \sum_{j=1}^I d_{ij}(\mathbf{u}^n) \right),$$

is obviously a conservative method in the sense of Definition 3.7. However, it is not an unconditionally positive scheme, because in general, we will be able to find a suitably large time step  $\Delta t > 0$  such that  $\mathbf{u}^{n+1} \not\geq \mathbf{0}$  even though  $\mathbf{u}^n$  is non-negative. The Patankar trick [29, 155] now weighs the destruction term by a certain factor that guarantees positivity but leads to an implicit formulation. More precisely, the Patankar-Euler method is given by

$$u_i^{n+1} = u_i^n + \Delta t \left( \sum_{j=1}^I p_{ij}(\mathbf{u}^n) - \sum_{j=1}^I d_{ij}(\mathbf{u}^n) \frac{u_i^{n+1}}{u_i^n} \right),$$

where we need the quantities  $\frac{d_{ij}(\mathbf{u}^n)}{u_i^n}$  to be well-defined as mentioned above.

We may rearrange the Patankar-Euler scheme into the form

$$u_i^{n+1} = \frac{u_i^n + \Delta t \sum p_{ij}(\mathbf{u}^n)}{1 + \Delta t \sum d_{ij}(\mathbf{u}^n)(u_i^n)^{-1}}$$

and hence obtain positivity as  $p_{ij}, d_{ij} \geq 0$ . However, the Patankar-Euler method is obviously not conservative. Therefore, the modified Patankar-Euler scheme by Burchard et al. [29] also weights the production term in an equivalent manner and is given by

$$u_i^{n+1} = u_i^n + \Delta t \left( \sum_{j=1}^I p_{ij}(\mathbf{u}^n) \frac{u_j^{n+1}}{u_j^n} - \sum_{j=1}^I d_{ij}(\mathbf{u}^n) \frac{u_i^{n+1}}{u_i^n} \right). \quad (3.55)$$

Due to the balance of weighted production and destruction terms, one can now prove both unconditional positivity and conservativity for this scheme as well as first order accuracy, see [29].

In addition, a modified Patankar scheme of second order based on Heun's predictor corrector method was constructed in [29]. It has the form

$$\begin{aligned} u_i^{(2)} &= u_i^n + \Delta t \left( \sum_{j=1}^I p_{ij}(\mathbf{u}^n) \frac{u_j^{(2)}}{u_j^n} - \sum_{j=1}^I d_{ij}(\mathbf{u}^n) \frac{u_i^{(2)}}{u_i^n} \right), \\ u_i^{n+1} &= u_i^n + \frac{\Delta t}{2} \left( \sum_{j=1}^I \left( p_{ij}(\mathbf{u}^n) + p_{ij}(\mathbf{u}^{(2)}) \right) \frac{u_j^{n+1}}{u_j^{(2)}} - \sum_{j=1}^I \left( d_{ij}(\mathbf{u}^n) + d_{ij}(\mathbf{u}^{(2)}) \right) \frac{u_i^{n+1}}{u_i^{(2)}} \right), \end{aligned} \quad (3.56)$$

and thus requires the solution of two linear systems of the size  $I \times I$ . In order to avoid confusion with multirate partitioned Runge-Kutta schemes, we will denote the above modified Patankar scheme by mPaRK2 instead of using the original identifier MPRK2. As shown in [29], the mPaRK2 scheme is second order accurate with respect to the local discretization error, conservative and unconditionally positive.

More recently, further two-stage second order modified Patankar Runge-Kutta schemes have been constructed by Kopecz and Meister in [98]. While in the early years of Patankar-Runge-Kutta schemes, higher than second order accuracy seemed unattainable, four-stage third order modified Patankar-Runge-Kutta schemes have been designed in [100]. However, in [99], the same authors were able to prove that it is impossible to construct three stage third-order modified Patankar-Runge-Kutta schemes when taking Patankar-weight denominators which are products of powers of previous stage values.

In the following Section 3.4.2, we will discuss the application of modified Patankar-Runge-Kutta schemes to system of ODEs which result from the space discretization of certain PDEs in production-destruction form. In addition, enforcing linear stability may result in inadvertently small time steps of explicit schemes applied to stiff problems. Therefore, an implicit modified Patankar-Runge-Kutta scheme is constructed in Section 3.4.3.

### 3.4.2 Quasi-linear production-destruction equations arising from PDEs

While many interesting biochemical reactions fit into the framework of ordinary differential equations in production-destruction form, it also includes certain space-discretized partial differential equations, e.g. the heat equation discretized by second-order differences and the first-order upwind-discretized advection equation.

The latter examples are quasi-linear ODEs which may be put into a vectorized production-destruction form

$$\mathbf{u}' = \tilde{\mathbf{P}}(\mathbf{u})\mathbf{u} - \tilde{\mathbf{Q}}(\mathbf{u})\mathbf{u}, \quad (3.57)$$

with matrix-valued functions  $\tilde{\mathbf{P}}, \tilde{\mathbf{Q}} : \mathbb{R}^I \rightarrow \mathbb{R}^{I \times I}$  such that

$$\tilde{\mathbf{P}}(\mathbf{u}) = (\tilde{p}_{ij}(\mathbf{u})) \geq \mathbf{0} \text{ and } \tilde{\mathbf{Q}}(\mathbf{u}) = \text{diag}(\tilde{q}_i(\mathbf{u})) \geq \mathbf{0}$$

for all  $\mathbf{u} \in \mathbb{R}^I$ . Thus, the component-wise formulation of the above system is given by

$$u'_i = \left( \sum_{j=1}^I \tilde{p}_{ij}(u) \cdot u_j \right) - \tilde{q}_i(u) \cdot u_i, \quad (3.58)$$

for  $i = 1, 2, \dots, I$ . Usually, we also have  $\tilde{p}_{ii}(\mathbf{u}) = 0$  for all  $\mathbf{u} \in \mathbb{R}^I$  and some mass conservation property such as  $\sum_{j=1}^I \tilde{p}_{ji}(\mathbf{u}) = \tilde{q}_i(\mathbf{u})$ .

Hence, we may recover the production-destruction formulation of (3.52) and (3.53) from the quasi-linear case by setting  $P_i(\mathbf{u}) = \sum_{j=1}^I p_{ij}(\mathbf{u}) = \sum_{j=1}^I \tilde{p}_{ij}(\mathbf{u})\mathbf{u}$  and  $D_i(\mathbf{u}) = \tilde{q}_i(\mathbf{u})u_i$ .

Setting  $\mathbf{A}(\mathbf{u}) = \tilde{\mathbf{P}}(\mathbf{u}) - \tilde{\mathbf{Q}}(\mathbf{u})$ , the production-destruction form (3.57) is written more shortly in the standard quasi-linear form  $\mathbf{u}' = \mathbf{A}(\mathbf{u})\mathbf{u}$ .

Generally, numerical methods discretizing (3.57) are supposed to be positivity preserving, conservative and of sufficiently high order. The Patankar-type approaches discussed in Section 3.4.1 might therefore be applicable in this context. However, while positivity preservation and conservativity directly carry over from ODEs to PDEs, the issue of consistency and convergence is more subtle for PDEs. In the following, we hence take a closer look at the local discretization error of Patankar-type methods applied to specific systems arising from the classical linear PDEs of advection and heat conduction. We thus consider autonomous systems of ordinary differential equations of the form

$$\mathbf{u}'(t) = \mathbf{A}\mathbf{u}(t), \quad (3.59)$$

where  $\mathbf{A}$  is obtained by spatial discretization of the linear advection or linear diffusion equation supplemented by periodic boundary conditions on equidistant grid points  $j\Delta x$  in the domain  $\Omega = [0, 1]$  and  $\mathbf{u}(t) = (u(x_1, t), \dots, u(x_I, t))^T$  is the corresponding vector of nodal values at time  $t$  of a sufficiently smooth function  $u \in C^k(\Omega, [0, T])$ , where  $k$  is sufficiently large. More precisely, in the following, we will consider the upwind discretization of the linear advection equation with periodic boundary conditions given by

$$\mathbf{A} = \frac{1}{\Delta x} \begin{pmatrix} -1 & & & 1 \\ 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{pmatrix}, \quad (3.60)$$

and the central discretization of the linear heat equation with periodic boundary conditions given by

$$\mathbf{A} = \frac{1}{\Delta x^2} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix}. \quad (3.61)$$

A first instructive example for the inadequacy of naive Patankar approaches is the inconsistency of the Patankar-Euler method for the semi-discrete linear heat equation.

### Inconsistency of the Patankar-Euler method

The forward Euler method applied to (3.57) will obviously be positivity preserving if we have  $\mathbf{I} - \Delta t \tilde{\mathbf{Q}}(\mathbf{u}^n) \geq \mathbf{0}$ , but this requires a very severe time step restriction on  $\Delta t$  for stiff systems. To avoid time step restrictions due to positivity enforcement, the Patankar-Euler method already introduced in the previous Section 3.4.1 written for the quasi-linear form (3.57) of the production-destruction equations is given by

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \tilde{\mathbf{P}}(\mathbf{u}^n) \mathbf{u}^n - \Delta t \tilde{\mathbf{Q}}(\mathbf{u}^n) \mathbf{u}^{n+1}. \quad (3.62)$$

This method has been proven unconditionally positivity preserving but it is not mass conserving. In addition, while (3.62) is of order one in the ODE sense, consistency is lost for stiff problems such as the discretized heat equation. In fact, the semi-discrete linear heat equation in one space dimension

$$u'_i(t) = \frac{1}{\Delta x^2} (u_{i-1}(t) - 2u_i(t) + u_{i+1}(t)), \quad i = 1, 2, \dots, I, \quad (3.63)$$

with spatial periodicity, i.e.  $u_0(t) = u_I(t)$  and  $u_{I+1}(t) = u_1(t)$  fits in the form (3.57) with diagonal destruction matrix  $\tilde{\mathbf{Q}}(\mathbf{u}) = 2\Delta x^{-2}\mathbf{I}$ . The Patankar-Euler scheme (3.62), written out per component, now reads

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x^2} (u_{i-1}^n - 2u_i^n + u_{i+1}^n). \quad (3.64)$$

This scheme is unconditionally positivity preserving as shown before. In addition, it is easily verified that it is also unconditionally contractive in the maximum norm, i.e.  $\|\mathbf{u}^{n+1}\|_\infty \leq \|\mathbf{u}^n\|_\infty$  for arbitrary  $\Delta t$ . However, consistency and convergence need to be regarded in the PDE sense. Inserting exact PDE solution values in the scheme, we obtain

$$u(x_i, t^{n+1}) = u(x_i, t^n) + \frac{\Delta t}{\Delta x^2} (u(x_{i-1}, t^n) - 2u(x_i, t^{n+1}) + u(x_{i+1}, t^n)) + \Delta t \rho_i^n,$$

with local truncation errors  $\rho_i^n$ . Taylor development shows that for small  $\Delta t$  and  $\Delta x$  the leading term in these local truncation errors is given by

$$\rho_i^n = \frac{2}{\Delta x^2} (u(x_i, t^{n+1}) - u(x_i, t^n)) = \frac{2\Delta t}{\Delta x^2} u_t(x_i, t^n) + \mathcal{O}\left(\frac{\Delta t^2}{\Delta x^2}\right).$$

It follows that the scheme will only be convergent in case of a very severe time step restriction of  $\frac{\Delta t}{\Delta x^2} \rightarrow 0$ . This leads to an even more drastic time step reduction than the undesirable stability restriction of  $\frac{\Delta t}{\Delta x^2} \leq \frac{1}{2}$  for the explicit Euler method applied to the semi-discrete linear heat equation.

### Modifications of the Patankar-Euler scheme

In order to obtain an unconditionally positive and additionally mass conservative scheme for production-destruction equations, the modified Patankar-Euler method may be used. For quasi-linear equations in production-destruction form, this method is given by

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \tilde{\mathbf{P}}(\mathbf{u}^n) \mathbf{u}^{n+1} - \Delta t \tilde{\mathbf{Q}}(\mathbf{u}^n) \mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \mathbf{A}(\mathbf{u}^n) \mathbf{u}^{n+1}.$$

For linear problems (3.59) with constant matrix  $\mathbf{A}$ , such as semi-discretizations of the linear advection or the linear heat equation, this modified method now reduces to the implicit Euler method, so consistency in PDE sense is not a problem there.

Next, we will consider the second-order mPaRK2 method proposed in [29], see also (3.56). This scheme does not fit directly in the vector production-destruction formulation and thus has to be written per component, starting with the quasi-linear form  $u'_i = \sum_{j=1}^I a_{ij}(u) u_j$ ,  $i = 1, 2, \dots, I$ . The mPaRK2 method is then based on the trapezoidal rule with an Euler-type prediction to provide the internal stage value  $v_i^{n+1} \approx u_i(t^{n+1})$  and reads

$$\begin{aligned} v_i^{n+1} &= u_i^n + \Delta t \sum_j a_{ij}(u^n) v_j^{n+1} \\ u_i^{n+1} &= u_i^n + \frac{1}{2} \Delta t \sum_j \left( a_{ij}(u^n) \frac{u_j^n}{v_j^{n+1}} u_j^{n+1} + a_{ij}(v^{n+1}) u_j^{n+1} \right). \end{aligned} \tag{3.65}$$

As shown in [29], the order of this scheme in the sense of ordinary differential equations is two. In general, it is unknown whether there will be order reduction for stiff problems, in particular for semi-discrete problems obtained from PDEs after space discretization. Regarding the local discretization error, in the following, we will prove consistency of the order  $\mathcal{O}(\Delta t^3)$  for sufficiently smooth exact solutions which are strictly positive. However, in the case that the exact solution vanishes at discrete points, the order of the local discretization error is reduced.

### Error recursions for semi-discrete linear advection and linear diffusion

We will study error recursions of the above mPaRK2 scheme (3.65) when applied to linear problems with constant coefficients. These are naturally non-linear for this method, even for linear equations. For a linear problem of the form (3.59) we will first write (3.65) in vector form by introducing the diagonal matrix  $\mathbf{W}^n = \text{diag}(u_i^n/v_i^{n+1})$ . Then (3.65) can be written compactly as

$$\begin{aligned} \mathbf{v}^{n+1} &= \mathbf{u}^n + \Delta t \mathbf{A} \mathbf{v}^{n+1} \\ \mathbf{u}^{n+1} &= \mathbf{u}^n + \frac{1}{2} \Delta t \mathbf{A} (\mathbf{W}^n + \mathbf{I}) \mathbf{u}^{n+1}. \end{aligned} \tag{3.66}$$

Along with this, we also consider the scheme with the exact solution inserted,

$$\begin{aligned}\bar{\mathbf{v}}^{n+1} &= \mathbf{u}(t^n) + \Delta t \mathbf{A} \bar{\mathbf{v}}^{n+1}, \\ \mathbf{u}(t^{n+1}) &= \mathbf{u}(t^n) + \frac{1}{2} \Delta t \mathbf{A} (\bar{\mathbf{W}}^n + \mathbf{I}) \mathbf{u}(t^{n+1}) + \boldsymbol{\rho}^n,\end{aligned}\tag{3.67}$$

where  $\bar{\mathbf{W}}^n = \text{diag}(u_i(t^n)/\bar{v}_i^{n+1})$  and  $\boldsymbol{\rho}^n = (\rho_i^n) \in \mathbb{R}^I$ . Subtraction of (3.66) from (3.67) gives a recursion for the global discretization errors  $\mathbf{e}^n = \mathbf{u}(t^n) - \mathbf{u}^n$  of the form

$$\mathbf{e}^{n+1} = \mathbf{R}^n \mathbf{e}^n + \mathbf{d}^n,$$

with amplification matrix and local errors given by

$$\mathbf{R}^n = \left( \mathbf{I} - \frac{1}{2} \Delta t \mathbf{A} (\bar{\mathbf{W}}^n + \mathbf{I}) \right)^{-1} \left( \mathbf{I} + \frac{1}{2} \Delta t \mathbf{A} \mathbf{G}^n \right),\tag{3.68}$$

$$\mathbf{d}^n = \left( \mathbf{I} - \frac{1}{2} \Delta t \mathbf{A} (\bar{\mathbf{W}}^n + \mathbf{I}) \right)^{-1} \boldsymbol{\rho}^n,\tag{3.69}$$

where  $\mathbf{G}^n \in \mathbb{R}^{I \times I}$  has the form

$$\mathbf{G}^n = \text{diag}(u_i^{n+1}/\bar{v}_i^{n+1}) - \text{diag}((u_i^n u_i^{n+1})/(\bar{v}_i^{n+1} v_i^{n+1})) (\mathbf{I} - \Delta t \mathbf{A})^{-1}.$$

The difference between  $\boldsymbol{\rho}^n$  and its counterpart resulting from the implicit trapezoidal rule can now be determined from

$$\boldsymbol{\rho}^n = \mathbf{u}(t^{n+1}) - \mathbf{u}(t^n) - \frac{1}{2} \Delta t \mathbf{A} (\mathbf{u}(t^n) + \mathbf{u}(t^{n+1})) + \frac{1}{2} \Delta t \mathbf{A} (\mathbf{u}(t^n) - \bar{\mathbf{W}}^n \mathbf{u}(t^{n+1})).$$

We may split the above truncation error into  $\boldsymbol{\rho}^n = \hat{\boldsymbol{\rho}}^n + \tilde{\boldsymbol{\rho}}^n$ , where

$$\hat{\boldsymbol{\rho}}^n = \mathbf{u}(t^{n+1}) - \mathbf{u}(t^n) - \frac{1}{2} \Delta t \mathbf{A} (\mathbf{u}(t^n) + \mathbf{u}(t^{n+1}))$$

is the truncation error of the trapezoidal rule and

$$\begin{aligned}\tilde{\boldsymbol{\rho}}^n &= \frac{1}{2} \Delta t \mathbf{A} (\mathbf{u}(t^n) - \bar{\mathbf{W}}^n \mathbf{u}(t^{n+1})) = \frac{1}{2} \Delta t \mathbf{A} \left( \mathbf{u}(t^n) - \text{diag} \left( \frac{u_i(t^n)}{\bar{v}_i^{n+1}} \right) \mathbf{u}(t^{n+1}) \right) \\ &= \frac{1}{2} \Delta t \mathbf{A} \text{diag} \left( \frac{\bar{v}_i^{n+1} - u_i(t^{n+1})}{\bar{v}_i^{n+1}} \right) \mathbf{u}(t^n)\end{aligned}\tag{3.70}$$

represents the difference in truncation errors between the implicit trapezoidal rule and the mPaRK2 scheme.

Since the considered systems of quasi-linear ODEs stem from spatial discretization of PDEs on specific grids, the matrix  $\mathbf{A}$  appearing in (3.70) contains negative powers of the cell size  $\Delta x$ . In order to obtain consistency for  $\Delta t, \Delta x \rightarrow 0$  simultaneously, additional smoothness assumptions of the type  $\mathbf{A}^k \mathbf{u}(t^n) = \mathcal{O}(1)$  are needed for specific values of  $k$  depending on the order of consistency. In the following, we will accordingly assume sufficient smoothness of  $\mathbf{u}(t^n)$ .

Furthermore, if we assume  $\mathbf{u}(t^n) > 0$  then time stepping by the forward Euler scheme preserves strict positivity as will be shown in Lemma 3.11, therefore we have  $\bar{\mathbf{v}}^{n+1} > 0$ . Unfortunately,

this is not enough for the diagonal matrix in the formulation (3.70) of  $\tilde{\rho}^n$  to be bounded by some moderate constant. For this, additional assumptions are needed on the boundedness of  $\mathbf{u}(t^n)$  away from zero which are included in the following Theorem 3.9 on the truncation error for specific finite difference discretizations of the linear advection and the linear diffusion equation.

**Theorem 3.9.** *We consider the application of mPaRK2 to the system (3.59), where  $\mathbf{A}$  is obtained by spatial discretization of the linear advection equation (3.60) or the linear diffusion equation (3.61) and  $\mathbf{u}(t) = (u(x_1, t), \dots, u(x_I, t))^T$  is the vector of nodal values of a sufficiently smooth function. Furthermore, we assume that there exists a constant  $\delta > 0$  such that*

$$u(x, t^n) > \delta, \quad (3.71)$$

for any  $x \in [0, 1]$ . Then the truncation error of the mPaRK2 scheme fulfills

$$\rho^n \in \mathcal{O}((\Delta t)^3), \quad (3.72)$$

if the time step is chosen according to  $\Delta t = \mathcal{O}(\Delta x)$ .

Before proving the assertions of Theorem 3.9, we need the following auxiliary result.

**Lemma 3.10.** *Let  $\mathbf{A}$  be the upwind discretization (3.60) of the linear advection equation or the central discretization (3.61) of the linear diffusion equation. Then the matrix  $\mathbf{I} - \Delta t \mathbf{A}$  is invertible for any  $\Delta t, \Delta x > 0$ .*

Furthermore, let  $\bar{\mathbf{v}}^{n+1}$  denote the result of one implicit Euler step applied to the vector of nodal values  $\mathbf{u}(t^n)$  at time  $t^n \in [0, T]$  of a given function  $u : \Omega \times [0, T] \rightarrow \mathbb{R}$ , i.e.

$$\bar{\mathbf{v}}^{n+1} = (\mathbf{I} - \Delta t \mathbf{A})^{-1} \mathbf{u}(t^n). \quad (3.73)$$

Then  $\bar{\mathbf{v}}^{n+1}$  fulfills the estimate  $\mathbf{A}^l \bar{\mathbf{v}}^{n+1} = \mathcal{O}(1)$  for  $l = 0, 1, \dots, k$ , and we have the expansion

$$\bar{\mathbf{v}}^{n+1} = \mathbf{u}(t^n) + \Delta t \mathbf{A} \mathbf{u}(t^n) + \Delta t^2 \mathbf{A}^2 \mathbf{u}(t^n) + \dots + \Delta t^{k-1} \mathbf{A}^{k-1} \mathbf{u}(t^n) + \mathcal{O}(\Delta t^k), \quad (3.74)$$

if  $u(\cdot, t^n)$  is sufficiently smooth as a function defined on  $\Omega$ , i.e.  $u(\cdot, t^n) = C^{k+1}(\Omega)$  in the case that  $\mathbf{A}$  is given by (3.60) and  $u(\cdot, t^n) = C^{2k+1}(\Omega)$  in the case (3.61).

*Proof.* First, we note that due to the finite difference discretization on equidistant grids and periodic boundary conditions, the matrix  $\mathbf{A}$  is circulant and obviously, the same holds for the matrix  $\mathbf{I} - \Delta t \mathbf{A}$ . Furthermore, applying the Gershgorin theorem to the matrix  $\mathbf{I} - \Delta t \mathbf{A}$  with  $\mathbf{A}$  given either by (3.60) or by (3.61) shows that the eigenvalues  $\lambda_i$ ,  $i = 1, \dots, I$ , of  $\mathbf{I} - \Delta t \mathbf{A}$  fulfill  $|\lambda_i| \geq 1$ . Thus  $\mathbf{I} - \Delta t \mathbf{A}$  is invertible for any choice of discretization parameters  $\Delta t, \Delta x > 0$ .

Furthermore, the spectral radius of the inverse fulfills  $\rho((\mathbf{I} - \Delta t \mathbf{A})^{-1}) \leq 1$ . Since circulant matrices are normal, we have  $\|(\mathbf{I} - \Delta t \mathbf{A})^{-1}\|_2 = \max\{|\lambda_i^{-1}|\} \leq 1$  and thus  $(\mathbf{I} - \Delta t \mathbf{A})^{-1} = \mathcal{O}(1)$  and  $\bar{\mathbf{v}}^{n+1} = \mathcal{O}(1)$  for  $\Delta t, \Delta x \rightarrow 0$ .

Left multiplying (3.73) by  $\mathbf{A}^l$  now yields

$$\mathbf{A}^l \bar{\mathbf{v}}^{n+1} = \mathbf{A}^l (\mathbf{I} - \Delta t \mathbf{A})^{-1} \mathbf{u}(t^n) = (\mathbf{I} - \Delta t \mathbf{A})^{-1} \mathbf{A}^l \mathbf{u}(t^n) = \mathcal{O}(1) \quad (3.75)$$

for  $l = 1, \dots, k$ , since the preassigned smoothness assumptions enforce  $\mathbf{A}^l \mathbf{u}(t^n) = \mathcal{O}(1)$  for  $l \leq k$ .

Inserting the definition

$$\bar{\mathbf{v}}^{n+1} = \mathbf{u}(t^n) + \Delta t \mathbf{A} \bar{\mathbf{v}}^{n+1} \quad (3.76)$$

recursively into the right-hand side of (3.76) we obtain

$$\begin{aligned} \bar{\mathbf{v}}^{n+1} &= \mathbf{u}(t^n) + \Delta t \mathbf{A} (\mathbf{u}(t^n) + \Delta t \mathbf{A} \bar{\mathbf{v}}^{n+1}) \\ &= \mathbf{u}(t^n) + \Delta t \mathbf{A} \mathbf{u}(t^n) + \Delta t^2 \mathbf{A}^2 (\mathbf{u}(t^n) + \Delta t \mathbf{A} \bar{\mathbf{v}}^{n+1}) \\ &= \mathbf{u}(t^n) + \Delta t \mathbf{A} \mathbf{u}(t^n) + \Delta t^2 \mathbf{A}^2 \mathbf{u}(t^n) + \dots + \Delta t^{k-1} \mathbf{A}^{k-1} \mathbf{u}(t^n) + \Delta t^k \mathbf{A}^k \bar{\mathbf{v}}^{n+1} \end{aligned}$$

and bounding  $\mathbf{A}^k \bar{\mathbf{v}}^{n+1}$  by (3.75) yields the assertion (3.74).  $\square$

We are now ready to prove the main assertion on the truncation error of mPaRK2 applied to the semi-discretizations specified by (3.60) and (3.61).

*Proof of Theorem 3.9.* By the smoothness assumptions of Theorem 3.9, Taylor expansion of the exact solution  $\mathbf{u}(t^{n+1})$  yields

$$\begin{aligned} \mathbf{u}(t^{n+1}) &= \mathbf{u}(t^n) + \Delta t \mathbf{u}'(t^n) + \frac{\Delta t^2}{2} \mathbf{u}''(t^n) + \frac{\Delta t^3}{6} \mathbf{u}'''(t^n) + \mathcal{O}(\Delta t^4) \\ &= \mathbf{u}(t^n) + \Delta t \mathbf{A} \mathbf{u}(t^n) + \frac{\Delta t^2}{2} \mathbf{A}^2 \mathbf{u}(t^n) + \frac{\Delta t^3}{6} \mathbf{A}^3 \mathbf{u}(t^n) + \mathcal{O}(\Delta t^4), \end{aligned}$$

where (3.59) was used to replace the derivatives of  $\mathbf{u}(t)$  by the application of  $\mathbf{A}$ .

From Lemma 3.10, it now follows that

$$\bar{\mathbf{v}}^{n+1} - \mathbf{u}(t^{n+1}) = \frac{\Delta t^2}{2} \mathbf{A}^2 \mathbf{u}(t^n) + \frac{5\Delta t^3}{6} \mathbf{A}^3 \mathbf{u}(t^n) + \mathcal{O}(\Delta t^4). \quad (3.77)$$

Inserting this expansion into the vector

$$\mathbf{w} = \text{diag} \left( \frac{\bar{v}_i^{n+1} - u_i(t^{n+1})}{\bar{v}_i^{n+1}} \right) \mathbf{u}(t^n) = \text{diag} \left( \frac{u_i(t^n)}{\bar{v}_i^{n+1}} \right) (\bar{\mathbf{v}}^{n+1} - \mathbf{u}(t^{n+1}))$$

occurring in the formulation (3.70) of  $\tilde{\rho}^n$ , we hence obtain

$$\mathbf{A} \mathbf{w} = \frac{\Delta t^2}{2} \mathbf{A} \text{diag} \left( \frac{u_i(t^n)}{\bar{v}_i^{n+1}} \right) \left( \mathbf{A}^2 \mathbf{u}(t^n) + \frac{5\Delta t}{3} \mathbf{A}^3 \mathbf{u}(t^n) + \mathcal{O}(\Delta t^2) \right). \quad (3.78)$$

We will first consider the case of upwind discretization of the linear advection equation where  $\mathbf{A}$  is given by (3.60). As common in the analysis of finite difference schemes, by Taylor expansion in space of the function  $u(\cdot, t^n)$  we have

$$\mathbf{A}^2 \mathbf{u}(t^n) = \mathbf{u}_{xx}(t^n) + \mathcal{O}(\Delta x),$$

where the vector of second spatial derivatives  $\mathbf{u}_{xx}(t^n) = (u_{xx}(x_1, t^n), \dots, u_{xx}(x_I, t^n))^T$ . In addition, the smoothness assumptions yield  $\mathbf{A}^3\mathbf{u}(t^n) = \mathcal{O}(1)$ . For the components of the vector  $\mathbf{Aw}$  given in (3.78), we thus have

$$\begin{aligned} (\mathbf{Aw})_i &= \frac{w_{i-1} - w_i}{\Delta x} = \frac{1}{\Delta x} \frac{(\bar{v}_{i-1}^{n+1} - u_{i-1}(t^{n+1})) u_{i-1}(t^n) \bar{v}_i^{n+1} - (\bar{v}_i^{n+1} - u_i(t^{n+1})) u_i(t^n) \bar{v}_{i-1}^{n+1}}{\bar{v}_{i-1}^{n+1} \bar{v}_i^{n+1}} \\ &= \frac{\Delta t^2}{2\Delta x} \frac{u_{xx}(x_{i-1}, t^n) u_{i-1}(t^n) \bar{v}_i^{n+1} - u_{xx}(x_i, t^n) u_i(t^n) \bar{v}_{i-1}^{n+1} + \mathcal{O}(\Delta x) + \mathcal{O}(\Delta t)}{\bar{v}_{i-1}^{n+1} \bar{v}_i^{n+1}} \\ &= \frac{\Delta t^2}{2} \left( -\frac{u_{xxx}(x_i, t^n) u_{i-1}(t^n) u_i(t^n)}{\bar{v}_{i-1}^{n+1} \bar{v}_i^{n+1}} + \mathcal{O}(1) \right), \end{aligned}$$

where we used that the time step is chosen as  $\Delta t = \mathcal{O}(\Delta x)$ .

Furthermore, since  $\bar{\mathbf{v}}^{n+1} = \mathbf{u}(t^n) + \mathcal{O}(\Delta t)$  and  $\mathbf{u}(t^n) > \delta \mathbf{1}$  by assumption, we have the estimate  $(\bar{v}_i^{n+1} \bar{v}_{i-1}^{n+1})^{-1} = \mathcal{O}(1)$  and hence it holds that  $\mathbf{Aw} = \mathcal{O}(\Delta t^2)$ .

Finally, consistency of the classical trapezoidal rule for sufficiently smooth problems yields

$$\hat{\rho}^n = \mathcal{O}(\Delta t^3)$$

for its truncation error. Consequently, the assertion (3.72) of Theorem 3.9 in case of the upwind finite difference discretization of the linear advection equation now follows directly from (3.70), i.e.

$$\rho^n = \hat{\rho}^n + \tilde{\rho}^n = \mathcal{O}(\Delta t^3) + \frac{1}{2} \Delta t \mathbf{Aw} = \mathcal{O}(\Delta t^3).$$

Next, the case of linear diffusion discretized by the classical central finite difference scheme will be studied. For this case, we have

$$\mathbf{A}^2\mathbf{u}(t^n) = \mathbf{u}_{xxxx}(t^n) + \mathcal{O}(\Delta x^2), \quad \mathbf{A}^3\mathbf{u}(t^n) = \frac{\partial^6 \mathbf{u}(t^n)}{\partial x^6} + \mathcal{O}(\Delta x^2). \quad (3.79)$$

Therefore, the components of  $\mathbf{Aw}$  are given by

$$\begin{aligned} (\mathbf{Aw})_i &= \frac{w_{i-1} - 2w_i + w_{i+1}}{\Delta x^2} \\ &= \frac{\bar{v}_{i-1}^{n+1} - u_{i-1}(t^{n+1})}{\Delta x^2} \frac{u_{i-1}(t^n)}{\bar{v}_{i-1}^{n+1}} - 2 \frac{\bar{v}_i^{n+1} - u_i(t^{n+1})}{\Delta x^2} \frac{u_i(t^n)}{\bar{v}_i^{n+1}} + \frac{\bar{v}_{i+1}^{n+1} - u_{i+1}(t^{n+1})}{\Delta x^2} \frac{u_{i+1}(t^n)}{\bar{v}_{i+1}^{n+1}} \\ &= \tau_1 + \tau_2 + \tau_3, \end{aligned}$$

where the terms  $\tau_1, \tau_2, \tau_3$  are obtained using the expansion of  $\bar{\mathbf{v}}^{n+1} - \mathbf{u}(t^{n+1})$  in (3.77) as well as the estimates (3.79) for  $\mathbf{A}^2\mathbf{u}(t^n)$  and  $\mathbf{A}^3\mathbf{u}(t^n)$ , i.e.

$$\begin{aligned} \tau_1 &= \frac{\Delta t^2}{2\Delta x^2} \left( u_{xxxx}(x_{i-1}, t^n) \frac{u_{i-1}(t^n)}{\bar{v}_{i-1}^{n+1}} - 2u_{xxxx}(x_i, t^n) \frac{u_i(t^n)}{\bar{v}_i^{n+1}} + u_{xxxx}(x_{i+1}, t^n) \frac{u_{i+1}(t^n)}{\bar{v}_{i+1}^{n+1}} + \mathcal{O}(\Delta x^2) \right), \\ \tau_2 &= \frac{5\Delta t^3}{6\Delta x^2} \left( \frac{\partial^6 u(x_{i-1}, t^n)}{\partial x^6} \frac{u_{i-1}(t^n)}{\bar{v}_{i-1}^{n+1}} - 2 \frac{\partial^6 u(x_i, t^n)}{\partial x^6} \frac{u_i(t^n)}{\bar{v}_i^{n+1}} + \frac{\partial^6 u(x_{i+1}, t^n)}{\partial x^6} \frac{u_{i+1}(t^n)}{\bar{v}_{i+1}^{n+1}} + \mathcal{O}(\Delta x^2) \right), \\ \tau_3 &= \mathcal{O}\left(\frac{\Delta t^4}{\Delta x^2}\right) = \mathcal{O}(\Delta t^2). \end{aligned}$$

Hereby, we again used that the time step is chosen as  $\Delta t = \mathcal{O}(\Delta x)$  and that  $(\bar{\mathbf{v}}^{n+1})^{-1} = \mathcal{O}(1)$ .

In order to bound the terms  $\tau_1$  and  $\tau_2$ , we now reformulate the expressions

$$\mathbf{y} = \mathbf{A} \operatorname{diag}(\bar{v}_i^{-1}) \mathbf{s}, \quad (3.80)$$

contained therein, where  $\mathbf{s} = (s_1, \dots, s_I)^T$  with  $s_i = \frac{\partial^k u(x_i, t^n)}{\partial x^k} u_i(t^n)$  for  $k = 4, 6$  and we use the short notation  $\bar{\mathbf{v}} = \bar{\mathbf{v}}^{n+1}$ .

For the  $i$ -th component of the vector  $\mathbf{y}$  defined in (3.80), we have

$$\begin{aligned} y_i &= \frac{1}{\Delta x^2} \left( \frac{s_{i-1}}{\bar{v}_{i-1}} - 2 \frac{s_i}{\bar{v}_i} + \frac{s_{i+1}}{\bar{v}_{i+1}} \right) = \frac{1}{\Delta x^2} \left( \frac{s_{i-1} - s_i}{\bar{v}_{i-1}} + \frac{(\bar{v}_i - \bar{v}_{i-1})s_i}{\bar{v}_{i-1}\bar{v}_i} + \frac{s_{i+1} - s_i}{\bar{v}_{i+1}} + \frac{(\bar{v}_i - \bar{v}_{i+1})s_i}{\bar{v}_i\bar{v}_{i+1}} \right) \\ &= \frac{s_{x,i}(\bar{v}_{i-1} - \bar{v}_i) + \frac{\Delta x}{2} s_{xx,i}(\bar{v}_{i+1} - \bar{v}_{i-1}) + \mathcal{O}(\Delta x^2)}{\Delta x \bar{v}_{i-1}\bar{v}_{i+1}} + s_i \frac{(\bar{v}_i - \bar{v}_{i-1})\bar{v}_{i+1} + (\bar{v}_i - \bar{v}_{i+1})\bar{v}_{i-1}}{\Delta x^2 \bar{v}_{i-1}\bar{v}_i\bar{v}_{i+1}} \\ &= \frac{s_{x,i}}{\bar{v}_{i-1}\bar{v}_{i+1}} \frac{(\bar{v}_{i-1} - \bar{v}_i)}{\Delta x} + \mathcal{O}(1) + \frac{s_i}{\bar{v}_{i-1}\bar{v}_i\bar{v}_{i+1}} \left( \frac{(\bar{v}_i - \bar{v}_{i-1})}{\Delta x} \frac{(\bar{v}_{i+1} - \bar{v}_{i-1})}{\Delta x} - \bar{v}_{i-1} \frac{\bar{v}_{i-1} - 2\bar{v}_i + \bar{v}_{i+1}}{\Delta x^2} \right), \end{aligned}$$

using smoothness assumptions on the function  $s : \Omega \rightarrow \mathbb{R}$ ,  $s(x) = \frac{\partial^k u(x, t^n)}{\partial x^k} u(x, t^n)$  for  $k = 4, 6$  and the designations  $s_x$  and  $s_{xx}$  for the first and second derivative, respectively.

Using  $\bar{\mathbf{v}}^{n+1} = \mathbf{u}(t^n) + \Delta t \mathbf{A} \mathbf{u}(t^n) + \mathcal{O}(\Delta t^2)$ , choosing the time step as  $\Delta t = \mathcal{O}(\Delta x)$  and employing the short notation  $\mathbf{u} = \mathbf{u}(t^n)$ ,  $\mathbf{u}_x = (u_x(x_1, t^n), \dots, u_x(x_I, t^n))^T$ , we may then bound  $\mathbf{y}$  by

$$\begin{aligned} y_i &= \frac{s_{x,i} u_{x,i}}{\bar{v}_{i-1}\bar{v}_{i+1}} + \mathcal{O}(1) \\ &+ \frac{s_i}{\bar{v}_{i-1}\bar{v}_i\bar{v}_{i+1}} \left( 2(u_{x,i})^2 - \bar{v}_{i-1} \frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2} + \frac{\Delta t(u_{x,i-1} - 2u_{x,i} + u_{x,i+1})}{\Delta x^2} + \mathcal{O}(1) \right), \end{aligned}$$

which results in  $\mathbf{y} = \mathcal{O}(1)$  since by assumption, we have  $\bar{v}_i^{-1} = \mathcal{O}(1)$  for  $i = 1, \dots, I$ .

Substituting  $s_i = u_{xxxx}(x_i, t^n) u_i(t^n)$ ,  $i = 1, \dots, I$ , into the definition of  $\mathbf{y}$  in (3.80), we therefore obtain  $\tau_1 = \mathcal{O}(\Delta t^2)$  whereas  $s_i = \frac{\partial^6 u(x_i, t^n)}{\partial x^6} u_i(t^n)$  yields  $\tau_2 = \mathcal{O}(\Delta t^3)$ .

Therefore, we have  $\mathbf{A} \mathbf{w} = \mathcal{O}(\Delta t^2)$  and  $\boldsymbol{\rho}^n = \mathcal{O}(\Delta t^3)$  proving the assertion of Theorem 3.9 also in the case of the central finite difference discretization of the linear heat equation.  $\square$

## Numerical results

In the following, the assertions of Theorem 3.9 will be numerically demonstrated by applying the mPaRK2 scheme to the upwind finite difference semi-discretization of the linear advection equation specified by the matrix  $\mathbf{A}$  in (3.60) and the central semi-discretization of the linear diffusion equation with  $\mathbf{A}$  in (3.61). Both test cases are supplemented by smooth initial conditions. The purpose of these experiments is to highlight the effect of the positivity condition (3.71) on the consistency of the mPaRK2 scheme.

We compute the respective local discretization errors for the first time step of the mPaRK2 scheme with time step size  $\Delta t = 0.1\Delta x$  applied to the finite difference semi-discretizations on successively refined grids having  $I = 20 \cdot 2^m$  grid points, where  $m = 1, \dots, 7$ . Hereby, the

$I$	$u_0 = 0.1 + \sin^2(2\pi x)$		$u_0 = \sin^2(2\pi x)$	
	$err_2(\Delta t)$	EOc	$err_2(\Delta t)$	EOc
40	1.78e-06		3.48e-06	
80	2.27e-07	2.97	6.17e-07	2.49
160	2.85e-08	2.99	1.07e-07	2.52
320	3.57e-09	2.99	1.87e-08	2.52
640	4.46e-10	3.00	3.29e-09	2.52
1280	5.58e-11	3.00	5.79e-10	2.51
2560	6.97e-12	3.00	1.02e-10	2.50

Table 3.13: Local error  $err_2(\Delta t)$  after the first time step and corresponding experimental order of consistency for mPaRK2 applied to semi-discrete linear advection for two different initial conditions.

local discretization errors are measured using a reference solution  $\mathbf{u}_{ref}(t)$  obtained by applying the fourth order Gauss-Runge-Kutta method to the semi-discrete system. Thus, the spatial discretization error is completely neglected. More precisely, the local discretization error at the end of the first time step, i.e. for  $t = \Delta t$ , is computed as

$$err_2(\Delta t) = \sqrt{\Delta x} \|\mathbf{u}_{ref}(\Delta t) - \mathbf{u}(\Delta t)\|_2.$$

Furthermore, from the errors on two consecutive grids, the experimental order of consistency (EOc) is determined.

Table 3.13 lists the local discretization errors after one time step of the mPaRK2 scheme for the semi-discrete linear advection equation as well as the experimental order of consistency while Table 3.14 shows the corresponding results for the semi-discrete linear diffusion equation. In accordance with the designed order of convergence, this local error behaves as  $\mathcal{O}(\Delta x^3) = \mathcal{O}(\Delta t^3)$  for an initial solution  $u_0 = 0.1 + \sin^2(2\pi x)$  satisfying the stricter positivity condition with  $\delta = 0.1$  in (3.71). On the other hand, for an initial solution of  $u_0 = \sin^2(2\pi x)$ , Table 3.14 shows an order reduction to about  $\mathcal{O}(\Delta t^{2.5})$  for the semi-discrete linear advection equation and to about  $\mathcal{O}(\Delta t^{2.3})$  for the semi-discrete linear diffusion equation. These numerical results hence agree with the assertions of Theorem 3.9.

So far, these investigations show a local discretization error of order  $\mathcal{O}(\Delta t^3)$  for sufficiently smooth and positive solutions. However, we should remark that for a full convergence analysis, stability has to be proven as well. This necessitates boundedness of products of the amplification matrices  $\mathbf{R}^n$  defined in (3.68) which seems to be quite difficult to prove due to the non-linearity of the method.

### Connections to thin-layer approaches for shallow water flow

In coastal engineering and marine ecosystems, an important feature of shallow water flows is the alternating exposure and submerging of the seabed which is referred to as wetting and drying. One of the computational approaches dealing with common stability issues due to the moving shoreline and vanishing water depth are thin layer techniques enforcing a minimum water depth such that a thin layer of water is present throughout the computational

$I$	$u_0 = 0.1 + \sin^2(2\pi x)$		$u_0 = \sin^2(2\pi x)$	
	$err_2(\Delta t)$	EOc	$err_2(\Delta t)$	EOc
40	1.77e-03		2.18e-03	
80	3.57e-04	2.31	5.37e-04	2.02
160	5.74e-05	2.64	1.17e-04	2.20
320	8.13e-06	2.82	2.45e-05	2.25
640	1.08e-06	2.91	5.12e-06	2.26
1280	1.40e-07	2.95	1.07e-06	2.26
2560	1.78e-08	2.97	2.24e-07	2.25

Table 3.14: Local error  $err_2(\Delta t)$  after the first time step and corresponding experimental order of consistency for mPaRK2 applied to semi-discrete linear diffusion for two different initial conditions.

domain. In this context, Section 4.4 discusses a DG scheme constructed by Meister and Ortob in [132], simulating two-dimensional shallow water flow subject to wetting and drying applying the Patankar approach to an implicit time integration scheme. In fact, the positivity requirement in Theorem 3.9 perfectly agrees with thin-layer approaches, where the thin film of water retained also in regions marked as dry corresponds to the positivity requirement given in (3.71).

In order to illustrate a possible improvement of the presently considered mPaRK2 scheme, in this paragraph, we compare it to a slight modification which follows more closely the approach in [132]. This adapted method is based on a direct correction of the explicit part of the implicit trapezoidal rule and reads as

$$\mathbf{v}^{n+1/2} = \mathbf{u}^n + \frac{\Delta t}{2} \mathbf{A} \mathbf{u}^n, \quad (3.81)$$

$$\mathbf{u}^{n+1/2} = \mathbf{u}^n + \frac{1}{2} \Delta t \mathbf{A} \tilde{\mathbf{W}} \mathbf{u}^{n+1/2}, \quad (3.82)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^{n+1/2} + \frac{1}{2} \Delta t \mathbf{A} \mathbf{u}^{n+1}, \quad (3.83)$$

with  $\tilde{\mathbf{W}} = \text{diag} \left( u_i^n / \tilde{v}_i^{n+1/2} \right)$  determined by the correction  $\tilde{\mathbf{v}}^{n+1/2}$  to the quantity  $\mathbf{v}^{n+1/2}$  which may have negative components. More precisely,  $\tilde{v}_i^{n+1/2}$  is given by  $\tilde{v}_i^{n+1/2} = v_i^{n+1/2}$  if  $v_i^{n+1/2} > 0$  and  $\tilde{v}_i^{n+1/2} = u_i^n$  otherwise. We will denote this scheme by mPaRK2ex. Due to this switch in case of vanishing components, we cannot expect an overall second order of convergence as the update reduces to two steps of the implicit Euler scheme if  $\mathbf{v}^{n+1/2} = 0$ . However, for a test case of an advected wave mimicking wetting and drying, i.e. advection of the initial condition  $u_0 = 0.01 + \sin^4(\pi x)$ , this method behaves much better than mPaRK2 as shown in Fig. 3.10.

A comparison of the Patankar-type schemes is carried out for the upwind-discretized linear advection on 160 grid points using spatial periodicity up to a final time of  $T = 2$ . As shown on the left of Fig. 3.10, using a time step of  $\Delta t = 0.025$  corresponding to a Courant number of 4 does not exhibit significant differences of the schemes mPaRK2 and mPaRK2ex, also in comparison to the implicit trapezoidal rule. However, a larger time step of  $\Delta t = 0.0625$

corresponding to a Courant number of 10 shows the drawback of mPaRK2 on the right part of Fig. 3.10. While mPaRK2 does not account for the vanishing solution in the interval  $[0, 0.15]$  and the implicit trapezoidal rule clearly yields negative values, the modified scheme mPaRK2ex seems to combine the best features of both methods. The solution is non-negative in the whole computational domain and very accurate in the almost dry regions. Hence, this method seems quite promising and should be further investigated, in particular with respect to its stability.

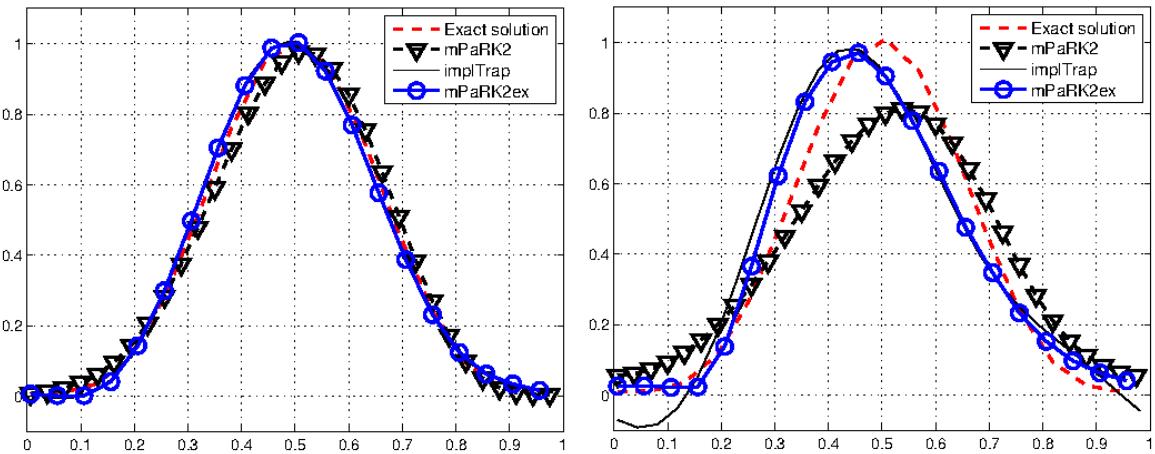


Figure 3.10: Linear transport of a wave using a Courant number of 4 (left) and 10 (right). Comparison of mPaRK2 to mPaRK2ex.

### 3.4.3 Applying the Patankar trick to an implicit RK scheme

Requiring non-negativity does not only restrict the time step sizes of classical explicit schemes but also of most implicit methods. In fact, even if the implicit scheme has a large stability region, unconditional positivity preservation is restricted to the implicit Euler method due to the order barrier by Bolley and Crouzeix [24].

A proof of unconditional positivity of the implicit Euler scheme can be found either in [84] or in [78], here we review the assertion of [84].

**Lemma 3.11.** *For a general nonlinear system of equations*

$$\mathbf{u}'(t) = \mathbf{g}(t, \mathbf{u})$$

*let the following conditions be fulfilled.*

1. *There exists  $\alpha = \alpha(\mathbf{u}) > 0$  such that*

$$\mathbf{u} + \Delta t \mathbf{g}(t, \mathbf{u}) \geq \mathbf{0} \text{ for all } t \geq 0, \mathbf{u} \geq \mathbf{0} \text{ and } \alpha \Delta t \leq 1.$$

2. *For any  $\mathbf{u} \geq \mathbf{0}, t \geq 0$  and  $\Delta t > 0$  the equation*

$$\mathbf{v} = \mathbf{u} + \Delta t \mathbf{g}(t, \mathbf{v})$$

*has a unique solution that depends continuously on  $\Delta t$  and  $\mathbf{v}$ .*

Then, the implicit Euler scheme  $\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \mathbf{g}(t, \mathbf{u}^{n+1})$  is unconditionally positive, i.e. if  $\mathbf{u}^n \geq \mathbf{0}$ , it yields  $\mathbf{u}^{n+1} \geq \mathbf{0}$  for any  $\Delta t > 0$ . In addition, we also have strict unconditional positivity preservation, i.e. if  $\mathbf{u}^n > \mathbf{0}$  then  $\mathbf{u}^{n+1} > \mathbf{0}$  for any  $\Delta t > 0$ .

*Proof.* Given  $t$  and  $\mathbf{u}$ , we denote the solution of the equation  $\mathbf{v} = \mathbf{u} + \tau \mathbf{g}(t, \mathbf{v})$  for a variable  $\tau$  by  $\mathbf{v}(\tau)$ . Since  $\mathbf{v}(\tau)$  is continuous with respect to  $\mathbf{u}$ , showing  $\mathbf{v}(\tau) > \mathbf{0}$  if  $\mathbf{u} > \mathbf{0}$  is sufficient. On the contrary, if we assume that  $v_i(\tau_0) = 0$  and  $v_j(\tau) > 0$  for all  $\tau \leq \tau_0$  for the other components with  $j \neq i$ , then

$$0 = v_i(\tau_0) = u_i + \tau_0 g_i(t, \mathbf{v}(\tau_0)).$$

However, since  $u_i > 0$ , this means  $g_i(t, \mathbf{v}(\tau_0)) < 0$ . Considering that we assumed  $v_i(\tau_0) = 0$ , we therefore have  $v_i(\tau_0) + \tau g_i(t, \mathbf{v}(\tau_0)) < 0$  for any  $\tau > 0$  which contradicts assumption 1.  $\square$

Naturally, the implicit Euler scheme is only first order accurate in terms of the local discretization error. Therefore, in order to enhance time accuracy of unconditionally positive implicit schemes, the Patankar approach has been applied to a higher order SDIRK scheme in [132]. In that work, the unconditionally positive *modified Patankar SDIRK3 (MPSDIRK3) scheme* has been constructed, where the underlying SDIRK scheme is the third order method of Cash [34], given by the Butcher array 3.15.

$\begin{array}{c ccc} \gamma & \gamma & & \\ \gamma + \delta & \delta & \gamma & \\ 1 & \alpha & \beta & \gamma \\ \hline & \alpha & \beta & \gamma \end{array}$	with	$\alpha = 1.2084966491760101,$ $\beta = -0.6443631706844691,$ $\gamma = 0.4358665215084580,$ $\delta = 0.2820667392457705.$
--	------	--

Table 3.15: SDIRK3 scheme by Cash.

The construction of this method is carried out as follows. As the implicit Euler scheme is unconditionally positive, we can safely apply the first stage of this SDIRK3 scheme, i.e.

$$u_i^{(1)} = u_i^n + \gamma \Delta t \left( P_i \left( \mathbf{u}^{(1)} \right) - D_i \left( \mathbf{u}^{(1)} \right) \right). \quad (3.84)$$

To obtain non-negativity for  $\mathbf{u}^{(1)}$  in (3.84) we only need  $\mathbf{u}^n \geq \mathbf{0}$ . In fact, as  $\delta < \gamma$ , we also have

$$u_i^n + \delta \Delta t \left( P_i \left( \mathbf{u}^{(1)} \right) - D_i \left( \mathbf{u}^{(1)} \right) \right) \geq 0, \quad \forall i \in I,$$

due to the non-negativity of  $\mathbf{u}^{(1)}$ . Therefore, unconditional positivity of the implicit Euler step leads to the fact that also the second stage,

$$u_i^{(2)} = u_i^n + \delta \Delta t \left( P_i \left( \mathbf{u}^{(1)} \right) - D_i \left( \mathbf{u}^{(1)} \right) \right) + \gamma \Delta t \left( P_i \left( \mathbf{u}^{(2)} \right) - D_i \left( \mathbf{u}^{(2)} \right) \right),$$

yields non-negative components  $\mathbf{u}^{(2)}$ . Hence, only the last stage needs to be modified. We set

$$\begin{aligned} u_i^{(3)} &= z_i^n + \gamma \Delta t \left( P_i \left( \mathbf{u}^{(3)} \right) - D_i \left( \mathbf{u}^{(3)} \right) \right), \\ u_i^{n+1} &= u_i^{(3)}, \end{aligned}$$

where  $z_i^n$  is a modification of the possibly negative state

$$\tilde{z}_i^n = u_i^n + \alpha \Delta t \left( P_i \left( \mathbf{u}^{(1)} \right) - D_i \left( \mathbf{u}^{(1)} \right) \right) + \beta \Delta t \left( P_i \left( \mathbf{u}^{(2)} \right) - D_i \left( \mathbf{u}^{(2)} \right) \right). \quad (3.85)$$

More precisely, we set

$$\begin{aligned} z_i^n &= u_i^n + \alpha \Delta t \left( \sum_{j=1}^I p_{ij} \left( \mathbf{u}^{(1)} \right) \frac{z_j^n}{\tilde{c}_j^{(1)}} - \sum_{j=1}^I d_{ij} \left( \mathbf{u}^{(1)} \right) \frac{z_i^n}{\tilde{c}_i^{(1)}} \right) \\ &\quad + \beta \Delta t \left( \sum_{j=1}^I p_{ij} \left( \mathbf{u}^{(2)} \right) \frac{z_i^n}{\tilde{c}_i^{(2)}} - \sum_{j=1}^I d_{ij} \left( \mathbf{u}^{(2)} \right) \frac{z_j^n}{\tilde{c}_j^{(2)}} \right), \end{aligned} \quad (3.86)$$

where for  $k = 1, 2$  we choose

$$\tilde{c}_j^{(k)} = \begin{cases} \tilde{z}_j^n, & \text{if } \tilde{z}_j^n > \epsilon, \\ u_j^{(k)}, & \text{otherwise.} \end{cases}$$

Note that in (3.86), since we have  $\beta < 0$ , the term  $d_{ij} \left( \mathbf{u}^{(2)} \right)$  now acts as a production term and  $p_{ij} \left( \mathbf{u}^{(2)} \right)$  as a destruction term. Therefore, also the corresponding weights have to be interchanged. In addition to well-defined quantities  $\frac{p_{ij} \left( \mathbf{u}^{(1)} \right)}{u_j^{(1)}} = \frac{d_{ji} \left( \mathbf{u}^{(1)} \right)}{u_j^{(1)}}$  we now also need

the quantities  $\frac{p_{ij} \left( \mathbf{u}^{(2)} \right)}{u_i^{(2)}} = \frac{d_{ji} \left( \mathbf{u}^{(2)} \right)}{u_i^{(2)}}$  to be well-defined. This latter requirement is the fulfilled if either  $u_i > 0$  throughout the computation or if all sequences  $\frac{p_{ij} \left( \mathbf{u} \right)}{u_i}$  with  $u_i \rightarrow 0$  are convergent.

The corrected vector  $\mathbf{z}^n$  is hence obtained as the solution of a linear system  $\mathbf{A}\mathbf{z}^n = \mathbf{u}^n$ , where the matrix  $\mathbf{A}$  has entries

$$\begin{aligned} a_{ii} &= 1 + \alpha \Delta t \sum_{j=1}^I \frac{d_{ij} \left( \mathbf{u}^{(1)} \right)}{\tilde{c}_i^{(1)}} - \beta \Delta t \sum_{j=1}^I \frac{p_{ij} \left( \mathbf{u}^{(2)} \right)}{\tilde{c}_i^{(2)}} \geq 1, \\ a_{ij} &= -\alpha \Delta t \sum_{j=1}^I \frac{p_{ij} \left( \mathbf{u}^{(1)} \right)}{\tilde{c}_j^{(1)}} + \beta \Delta t \sum_{j=1}^I \frac{d_{ij} \left( \mathbf{u}^{(2)} \right)}{\tilde{c}_j^{(2)}} \leq 0, \quad \text{for } i \neq j. \end{aligned}$$

Under the assumption that  $P_i, D_i$  are sufficiently smooth functions for  $i = 1, \dots, I$ , we now have the following theorem concerning positivity, conservativity and consistency of the MPSDIRK3 method.

**Theorem 3.12.** *The MPSDIRK3 scheme is conservative, unconditionally positive and first order accurate in terms of the local truncation error.*

**Proof:** In order to prove the conservativity of the MPSDIRK3 scheme, it is sufficient to show that the construction of  $\mathbf{z}^n$  is conservative, i.e.  $\sum_{i=1}^I z_i^n = \sum_{i=1}^I c_i^n$ . However, this is easily

seen from the implicit definition given in (3.86), as it yields

$$\begin{aligned}
\sum_{i=1}^I (z_i^n - u_i^n) &= \alpha \Delta t \sum_{i=1}^I \left( \sum_{j=1}^I p_{ij} (\mathbf{u}^{(1)}) \frac{z_j^n}{\tilde{c}_j^{(1)}} - \sum_{j=1}^I d_{ij} (\mathbf{u}^{(1)}) \frac{z_i^n}{\tilde{c}_i^{(1)}} \right) \\
&\quad + \beta \Delta t \sum_{i=1}^I \left( \sum_{j=1}^I p_{ij} (\mathbf{u}^{(2)}) \frac{z_i^n}{\tilde{c}_i^{(2)}} - \sum_{j=1}^I d_{ij} (\mathbf{u}^{(2)}) \frac{z_j^n}{\tilde{c}_j^{(2)}} \right) \\
&= \alpha \Delta t \left( \sum_{i,j=1}^I p_{ij} (\mathbf{u}^{(1)}) \frac{z_j^n}{\tilde{c}_j^{(1)}} - \sum_{i,j=1}^I p_{ji} (\mathbf{u}^{(1)}) \frac{z_i^n}{\tilde{c}_i^{(1)}} \right) \\
&\quad + \beta \Delta t \left( \sum_{i,j=1}^I p_{ij} (\mathbf{u}^{(2)}) \frac{z_i^n}{\tilde{c}_i^{(2)}} - \sum_{i,j=1}^I p_{ji} (\mathbf{u}^{(2)}) \frac{z_j^n}{\tilde{c}_j^{(2)}} \right) = 0.
\end{aligned}$$

Regarding positivity, by using the same arguments as in the proof of [29, Theorem 3.5], we obtain that  $\mathbf{A}$  is an M-Matrix. This yields unconditional positivity as for any time step size  $\Delta t > 0$ , we have  $\mathbf{z}^n \geq 0$  if  $\mathbf{u}^n \geq 0$ .

Furthermore, by using the same arguments as in [29, Lemma 3.10], we can show that the entries of the matrix  $\mathbf{A}^{-1} = (\tilde{a}_{ij})$  satisfy  $0 \leq \tilde{a}_{ij} \leq 1$  independent of the time step size. Therefore, we have  $z_i^n = \sum_{j=1}^I \tilde{a}_{ij} u_j^n = \mathcal{O}(1)$  for  $\Delta t \rightarrow 0$ . Due to the boundedness of the quantities  $\frac{p_{ij}(\mathbf{u}^{(1)})}{\tilde{c}_j^{(1)}} = \frac{d_{ji}(\mathbf{u}^{(1)})}{\tilde{c}_j^{(1)}}$ ,  $\frac{p_{ij}(\mathbf{u}^{(2)})}{\tilde{c}_i^{(2)}} = \frac{d_{ji}(\mathbf{u}^{(2)})}{\tilde{c}_i^{(2)}}$  and  $z_i^n$ , equation (3.86) then yields  $z_i^n - u_i^n = \mathcal{O}(\Delta t)$  for  $i = 1, \dots, I$ . As  $u_i^{(k)} = u_i^n + \mathcal{O}(\Delta t)$ , for  $k = 1, 2$ , and  $\tilde{z}_i^n = u_i^n + \mathcal{O}(\Delta t)$ , we immediately obtain

$$z_i^n - \tilde{c}_i^{(k)} = \mathcal{O}(\Delta t). \quad (3.87)$$

For the difference  $\mathbf{z} - \tilde{\mathbf{z}}$  we now have

$$\begin{aligned}
z_i^n - \tilde{z}_i^n &= \alpha \Delta t \left( \sum_{j=1}^I \frac{p_{ij} (\mathbf{u}^{(1)})}{\tilde{c}_j^{(1)}} (z_j^n - \tilde{c}_j^{(1)}) - \sum_{j=1}^I \frac{d_{ij} (\mathbf{u}^{(1)})}{\tilde{c}_i^{(1)}} (z_i^n - \tilde{c}_i^{(1)}) \right) \\
&\quad + \beta \Delta t \left( \sum_{j=1}^I \frac{p_{ij} (\mathbf{u}^{(2)})}{\tilde{c}_i^{(2)}} (z_i^n - \tilde{c}_i^{(2)}) - \sum_{j=1}^I \frac{d_{ij} (\mathbf{u}^{(2)})}{\tilde{c}_j^{(2)}} (z_j^n - \tilde{c}_j^{(2)}) \right) = \mathcal{O}(\Delta t^2).
\end{aligned} \quad (3.88)$$

Since the MPSDIRK3 scheme is a perturbation of the third order SDIRK3 scheme

$$\begin{aligned}
\tilde{c}_i^{(3)} &= \tilde{z}_i^n + \gamma \Delta t \left( P_i (\tilde{\mathbf{u}}^{(3)}) - D_i (\tilde{\mathbf{u}}^{(3)}) \right), \\
\tilde{c}_i^{n+1} &= \tilde{c}_i^{(3)},
\end{aligned}$$

with unperturbed state  $\tilde{\mathbf{z}}^n$  defined in (3.85), we have

$$u_i^{n+1} - u_i(t^{n+1}) = u_i^{n+1} - \tilde{c}_i^{n+1} + \tilde{c}_i^{n+1} - u_i(t^{n+1}) = u_i^{n+1} - \tilde{c}_i^{n+1} + \mathcal{O}(\Delta t^4).$$

Thus the MPSDIRK3 scheme can be at most of third order. Now, due to Lipschitz continuity, we directly see

$$\begin{aligned} u_i^{n+1} - \tilde{c}_i^{n+1} &= u_i^{(3)} - \tilde{c}_i^{(3)} = \underbrace{z_i^n - \tilde{z}_i^n}_{=\mathcal{O}(\Delta t^2)} + \gamma \Delta t \left( P_i \left( \mathbf{u}^{(3)} \right) - P_i \left( \tilde{\mathbf{u}}^{(3)} \right) - D_i \left( \mathbf{u}^{(3)} \right) + D_i \left( \tilde{\mathbf{u}}^{(3)} \right) \right) \\ &= \mathcal{O}(\Delta t). \end{aligned}$$

For smooth  $P_i, D_i$  we furthermore have

$$\begin{aligned} u_i^{n+1} - \tilde{c}_i^{n+1} &= u_i^{(3)} - \tilde{c}_i^{(3)} \\ &= z_i^n - \tilde{z}_i^n + \gamma \Delta t \left( \sum_{j=1}^I \frac{\partial(P_i - D_i)}{\partial u_j} (\mathbf{u}^{(3)}) \underbrace{(u_j^{(3)} - \tilde{c}_j^{(3)})}_{=\mathcal{O}(\Delta t)} + \mathcal{O} \left( (u_j^{(3)} - \tilde{c}_j^{(3)})^2 \right) \right) \\ &= \mathcal{O}(\Delta t^2). \end{aligned} \tag{3.89}$$

Hence, for the complete MPSDIRK3 scheme we obtain  $\mathbf{u}^{n+1} = \mathbf{u}(t^{n+1}) + \mathcal{O}(\Delta t^2)$ .  $\square$

Though the global accuracy of the scheme, considering all components of  $\mathbf{u}$ , is restricted to at most first order, components that have moderate values not too close to zero may still be approximated with higher order accuracy, as we will see next. We consider a given component  $u_i$  and assume that the two following conditions hold:

(C1)  $\forall j \in \{1, \dots, I\}$ :

1. either  $p_{ij}(\mathbf{u}^{(1)}) = 0$  or  $\tilde{c}_j^{(1)} = \tilde{z}_j^n$  and
2. either  $d_{ij}(\mathbf{u}^{(2)}) = 0$  or  $\tilde{c}_j^{(2)} = \tilde{z}_j^n$ ,

(C2) 

1. either  $d_{ij}(\mathbf{u}^{(1)}) = 0 \forall j \in \{1, \dots, I\}$  or  $\tilde{c}_i^{(1)} = \tilde{z}_i^n$  and
2. either  $p_{ij}(\mathbf{u}^{(2)}) = 0 \forall j \in \{1, \dots, I\}$  or  $\tilde{c}_i^{(2)} = \tilde{z}_i^n$ .

Then we obtain  $z_i^n - \tilde{z}_i^n = \mathcal{O}(\Delta t^3)$  using equation (3.88) in the proof of Theorem 3.12. For the given component  $u_i$  we then have  $u_i^{n+1} - u_i(t^{n+1}) = \mathcal{O}(\Delta t^3)$  due to the first equality in (3.89).

If for a given component  $u_i$ , in addition to the conditions (C1) and (C2), we have

(C3)  $\forall j \in \{1, \dots, I\} : \begin{cases} \text{either (C1) and (C2) also hold for } u_j, \\ \text{or } p_{ij}(\mathbf{u}^{(1)}) = d_{ij}(\mathbf{u}^{(2)}) = 0 \text{ and } \frac{\partial(P_i - D_i)}{\partial u_j}(\mathbf{u}) \equiv 0, \end{cases}$

iterating the previous argument yields  $z_i^n - \tilde{z}_i^n = \mathcal{O}(\Delta t^4)$  as well as  $u_i^{n+1} - u_i(t^{n+1}) = \mathcal{O}(\Delta t^4)$ .

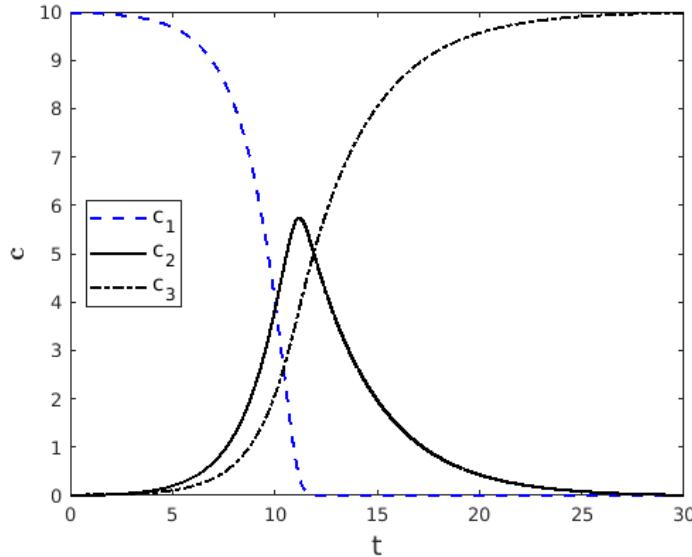


Figure 3.11: High-resolution approximation to production-destruction equation (3.90).

### Convergence study for a nonlinear ODE of production-destruction type

In order to study the MPSDIRK3 scheme in the context of ordinary differential equations in production-destruction form (3.52), we carry out a numerical convergence study for a nonlinear geobiochemical model also considered in [29]. This system of equations for positive constituents  $\mathbf{c}(t)$  is given by

$$\begin{aligned} c'_1 &= -\frac{c_2 c_1}{c_1 + 1}, \\ c'_2 &= \frac{c_2 c_1}{c_1 + 1} - ac_2, \\ c'_3 &= ac_2, \end{aligned} \tag{3.90}$$

with  $a = 0.3$ . For this system, we have  $d_{12} = p_{21} = -\frac{c_2 c_1}{c_1 + 1}$  and  $d_{23} = p_{32} = ac_2$ . Given initial data  $\mathbf{c}(0) = (9.98, 0.01, 0.01)^T$ , a reference solution for the time  $t = 30$  is depicted in Figure 3.11. This reference solution was obtained using the underlying SDIRK3 scheme by Cash with a very small time step of  $\Delta t = 0.001$ .

As in [29], given the reference solution  $\mathbf{c}(t)$  and numerical approximations  $\mathbf{c}^n$  for  $n = 1, \dots, N_{\Delta t}$ , we measure the truncation errors  $E(\Delta t)$  by

$$E(\Delta t) = \sqrt{\frac{1}{N_{\Delta t}} \sum_{n=1}^{N_{\Delta t}} (c_1(n\Delta t) - c_1^n)^2} \Bigg/ \left( \frac{1}{N_{\Delta t}} \sum_{n=1}^{N_{\Delta t}} c_1(n\Delta t) \right).$$

For the MPSDIRK3 scheme as well as the usual SDIRK3 method the truncation errors are given in Table 3.16. While the error histories of both schemes coincide for time steps  $\Delta t \leq 0.3$ , we observe an extremely large error for SDIRK3 with time step  $\Delta t = 1.2$  whereas the

corresponding error for MPSDIRK3 is moderate. The SDIRK3 solution actually blows up in this case due to the appearance of negative concentrations. For both schemes, Table 3.16 also lists the conservation errors at time  $t = 30$  given by

$$e_{cons}(\Delta t) = 10 - \sum_{i=1}^3 c_i^{N_{\Delta t}}.$$

These errors are due to the fact that we approximately solve the nonlinear systems occurring for each RK stage by Newton iteration. Obviously, for both schemes the conservation errors are negligible. Figure 3.12 again depicts the truncation errors versus time step size in double logarithmic scale. Here we clearly see the higher order of the SDIRK3 and the MPSDIRK3 scheme compared to the first-order implicit Euler method and the modified Patankar-Euler scheme as well as to the second-order mPaRK2 scheme.

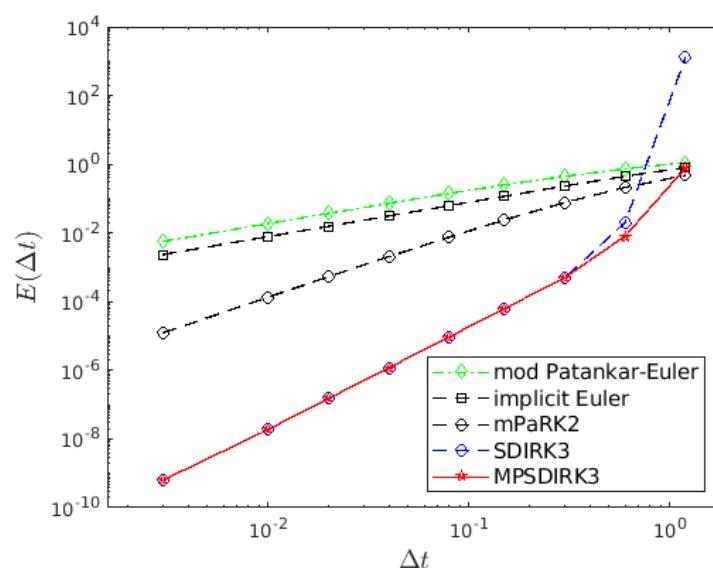
For small time steps we may as well use the SDIRK3 scheme instead of MPSDIRK3. The improvement caused by the MPSDIRK3 scheme shows in the range of moderate to large time steps. This fact is depicted in Figures 3.13 and 3.14. Figure 3.13 shows the blow-up of the SDIRK3 solution to equation (3.90) for a time step of  $\Delta t = 0.75$  which is due to a negative concentration of  $c_1$ . This is in contrast to the corresponding MPSDIRK3 solution for  $\Delta t = 0.75$  which is shown in Figure 3.14. Here the solution stays positive throughout the time interval. Although in the numerical solution shows an increase of  $c_1$  with an unphysical peak near  $t = 13$  as well as a corresponding decrease of  $c_2$ , the main features are well-represented on this coarse time-grid.

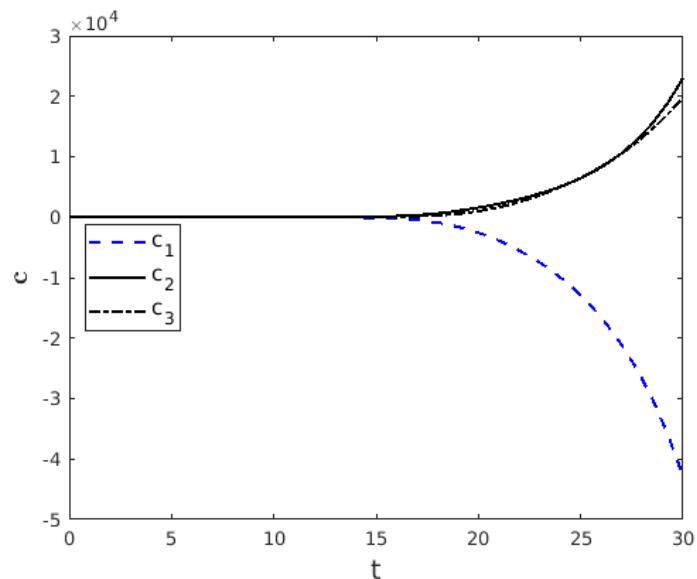
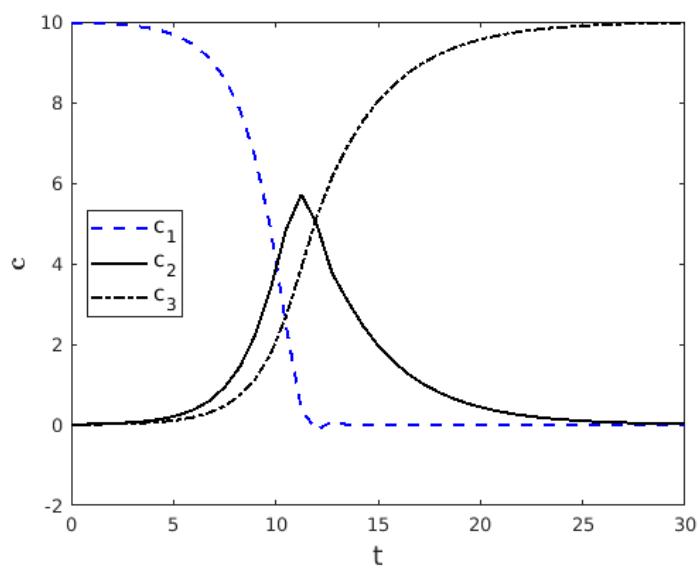
Although there may be examples where the numerical simulation can still be continued despite the computation of unphysical negative quantities, this is in general not the case in the context of shallow water flows which are considered in more detail in Chapter 4. In particular, the numerical fluxes employed within the spatial discretization of the shallow water equations needs to compute the square root of the water depth to approximately compute the characteristic speeds. The unconditional positivity of the MPSDIRK3 scheme thus poses a clear advantage since larger time steps of the implicit scheme may actually be employed in order to beat explicit time-stepping in terms of CPU time.

In the context of shallow water flows discretized by the discontinuous Galerkin method, a suitable production-destruction equation is specifically formulated by Meister and Ortob in [132], in order to account for ingoing and outgoing water flows which influence the cell-wise water volume. Applying the MPSDIRK3 scheme in this framework thereby guarantees non-negativity of the water height for any time step size while still preserving conservativity, as discussed in Section 4.4.

$\Delta t$	MPSDIRK3		SDIRK3	
	$E(\Delta t)$	$e_{cons}(\Delta t)$	$E(\Delta t)$	$e_{cons}(\Delta t)$
1.2	7.8093e-01	1.243e-14	1.3006e+03	1.091e-11
0.6	8.0174e-03	3.553e-15	2.0372e-02	-5.329e-15
0.3	4.9465e-04	3.553e-15	4.9465e-04	-7.105e-15
0.15	6.1675e-05	-1.066e-14	6.1675e-05	1.776e-14
8.0e-02	9.4199e-06	-7.105e-15	9.4199e-06	4.086e-14
4.0e-02	1.1856e-06	-1.066e-14	1.1856e-06	-3.908e-14
2.0e-02	1.4893e-07	1.865e-13	1.4893e-07	1.350e-13
1.0e-02	1.8759e-08	1.563e-13	1.8759e-08	1.599e-13
3.0e-03	6.2983e-10	7.105e-14	6.2983e-10	-3.908e-14

Table 3.16: Truncation and conservation errors for MPSDIRK3 and SDIRK3 scheme.

Figure 3.12: Truncation errors vs.  $\Delta t$  in double logarithmic scale.

Figure 3.13: Blow-up of SDIRK3 solution to (3.90) for  $\Delta t = 0.75$ .Figure 3.14: MPSDIRK3 solution to (3.90) for  $\Delta t = 0.75$ .



## Chapter 4

# Wetting and Drying Shallow Water Flows

### *Designing Positivity Preserving, Conservative and Well-balanced Schemes*

The study of ocean dynamics as well as the simulation of flows in rivers or lakes requires the simulation of water flow in diverse regions of varying fluid depth. Part of the considered regions may therefore be subject to alternating wetting and drying processes where the water depth may drastically change. In particular, the alternating exposure and submerging of the seabed is an important feature in coastal engineering and marine ecosystems.

Wetting and drying occurs on different time scales such as hours in the case of the tides or days in the case of storm surges. The situations relevant to wetting and drying include coastal regions of different characteristics, such as shores, embayments, tidal flats or estuaries. Capturing the fluid dynamics in these areas is significant in order to study and possibly predict singular phenomena such as storm surges or inundations. In addition, repetitive flooding and ebbing is vital to the local ecosystem. Hence, in modeling the occurring biological processes, the time evolution of the flooding and receding water front plays an important role. Furthermore, the periodic occurrence of wetting and drying due to the tides affects sediment transport. Consequently, the alternating run-up on beaches and dunes and the subsequent receding of the water front may cause coastal erosion.

The importance of capturing wetting and drying shallow water flows in coastal engineering and marine ecosystems is accompanied by challenges with respect to both the development of suitable mathematical models of the occurring processes and the construction of accurate and robust numerical methods. Wetting and drying in shallow regions of the flow challenges the numerical simulation in terms of robustness, efficiency, and the preservation of physical properties. More precisely, for a physically sound representation, desired numerical properties include positivity preservation with respect to the water depth, local and global mass conservation, well-balancedness with respect to lake at rest steady states, and avoidance of artificial pressure gradients.

### Organization of this chapter

Considering mathematical models, this chapter focuses on the depth-averaged shallow water (SW) equations which are commonly used in coastal areas. These equations are a valid model of water flow in the case that the horizontal length scale is significantly greater than the vertical one and the vertical velocity is comparatively small with respect to the horizontal velocity. Therefore, in Section 4.1, the most popular variants of the 2D shallow water equations are described. In particular, we discuss the different formulations with respect to their impact on well-balancedness of the numerical schemes for still water steady states.

Subsequently, the first part of this chapter presents a review on numerical methods for wetting and drying shallow water flows on fixed grids in two space dimensions based on finite volume and discontinuous Galerkin space discretization. Hereby, Section 4.2 discusses the general challenges faced by numerical methods for this application and Section 4.3 reviews the wetting and drying treatment of finite volume methods in Section 4.3.1 and discontinuous Galerkin schemes in Section 4.3.2. Section 4.4 deals with a Patankar approach developed for a production-destruction formulation describing the time evolution of the water height averages within a DG scheme on triangular grids. Thereby, a positivity preserving and well-balanced DG scheme is combined with implicit time integration. Due to the incorporation of the MPSDIRK3 scheme constructed in Section 3.4.3, for stiff problems, this implicit scheme can take full advantage of larger time steps and is therefore able to beat explicit time stepping in terms of CPU time.

## 4.1 The governing equations of shallow water flow

The SW equations are based on the assumption of a small vertical length scale compared to large horizontal ones and a hydrostatic pressure distribution. The derivation of these governing equations is based on depth-integration of the Navier-Stokes equations which removes the vertical velocity from the set of variables. As such, the SW equations simplify the reality but represent an important model in many scientific and engineering applications.

Near the wet/dry front, however, these assumptions are not fully valid. First, when the fluid depth vanishes, these equations become ill-posed. In addition, if the model includes bottom friction, e.g. in form of a Manning friction term with experimentally determined roughness coefficient, this again involves division by the fluid depth. Second, near the front the horizontal and vertical length scales become comparable. Indeed, the hydrostatic assumption may be violated in certain cases, such as Tsunami modeling which may require the inclusion of non-hydrostatic effects, as discussed in [31].

Nevertheless, on moderate scales as in estuaries or mud flats, the SW equations may capture the relevant dynamics quite efficiently and to a sufficient degree of accuracy. They can be used to provide realistic simulations of flows in rivers, lakes or coastal areas. If the bottom topography is assumed to be constant with respect to time but non-flat in space, the SW equations in *conservative form* in two space dimensions, are given by

$$\begin{aligned} \frac{\partial H}{\partial t} + \frac{\partial Hv_1}{\partial x} + \frac{\partial Hv_2}{\partial y} &= 0, \\ \frac{\partial Hv_1}{\partial t} + \frac{\partial(Hv_1^2 + \frac{g}{2}H^2)}{\partial x} + \frac{\partial Hv_1v_2}{\partial y} - fHv_2 &= -gH\frac{\partial b}{\partial x} + \frac{\tau_x^s - \tau_x^b}{\rho}, \\ \frac{\partial Hv_2}{\partial t} + \frac{\partial Hv_1v_2}{\partial x} + \frac{\partial(Hv_2^2 + \frac{g}{2}H^2)}{\partial y} + fHv_1 &= -gH\frac{\partial b}{\partial y} + \frac{\tau_y^s - \tau_y^b}{\rho}, \end{aligned} \quad (4.1)$$

where  $H$  is the water column height,  $\mathbf{v} = (v_1, v_2)^T$  is the fluid velocity vector,  $b$  is the bottom topography and  $g$  denotes the gravitational constant. Furthermore, Coriolis forces are included in this formulation, where  $f$  denotes the Coriolis parameter depending on the geographic latitude. Forces due to wind stress and bottom friction are contained, denoting the surface stresses by  $\tau_x^s, \tau_y^s$  and bottom stresses by  $\tau_x^b, \tau_y^b$ , while  $\rho$  is the density of water. Further influences, such as eddy viscosity, may be included within the momentum equations but have been neglected in the above formulation. The SW equations in conservative form can be rewritten more compactly as

$$\frac{\partial}{\partial t} \mathbf{U}(x, y, t) + \nabla \cdot \mathbf{F}(\mathbf{U}(x, y, t)) = \mathbf{S}(\mathbf{U}(x, y, t), x, y), \quad (4.2)$$

where the conservative variables are now collected in  $\mathbf{U} = (H, Hv_1, Hv_2)^T$ , while  $\mathbf{F}$  contains the fluxes, such that  $\nabla \cdot \mathbf{F} = \frac{\partial}{\partial x} \mathbf{F}_1 + \frac{\partial}{\partial y} \mathbf{F}_2$  with

$$\mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} Hv_1 \\ Hv_1^2 + \frac{g}{2}H^2 \\ Hv_1v_2 \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} Hv_2 \\ Hv_1v_2 \\ Hv_2^2 + \frac{g}{2}H^2 \end{pmatrix}, \quad (4.3)$$

and  $\mathbf{S}$  contains the sources, i.e.

$$\mathbf{S} = \begin{pmatrix} 0 \\ fHv_2 - gH\frac{\partial}{\partial x}b + \frac{\tau_x^s - \tau_x^b}{\rho} \\ -fHv_1 - gH\frac{\partial}{\partial y}b + \frac{\tau_y^s - \tau_y^b}{\rho} \end{pmatrix}.$$

Sometimes, the SW equations are rewritten in terms of the geopotential  $\varphi = gH$ .

When solving the conservative formulation of the SW equations, a central requirement is to satisfy the well-balanced property for still water stationary solutions, i.e. to maintain a lake at rest steady state solution. As already discussed in Section 1.5.1, this precise steady state is given by a constant sea surface elevation  $\eta = H + b = \text{const}$  and a zero velocity vector, i.e.  $\mathbf{v} = \mathbf{0}$ . When neglecting wind stresses, these steady states are obviously exact solutions of the analytical equations above, basically due to the fact that the net pressure forces vanish. More precisely, due to the constant sea surface, we have the following balance for the first of the momentum equations

$$\frac{g}{2} \frac{\partial H^2}{\partial x} + gH \frac{\partial}{\partial x} b = gH \frac{\partial}{\partial x} \eta = 0, \quad (4.4)$$

and an analogous one for the second momentum equation. Hence, in the analytical equations the pressure forces are split in two parts which attain non-zero values of opposite signs in case of non-constant bottom topography  $b$  and thus cancel out. Also in the context of wetting and drying, one has to ensure that the numerical scheme satisfies this analytical property as well, at least within machine-precision. With respect to wetting and drying methods, simultaneously satisfying both the well-balanced property and non-negativity of the water column height is not trivial, see e.g. Xing et al. [214]. This issue may not arise for slightly modified formulations such as the reformulation

$$\frac{\partial Hv_1}{\partial t} + \frac{\partial Hv_1^2}{\partial x} + \frac{\partial Hv_1v_2}{\partial y} - fHv_2 = -gH\frac{\partial \eta}{\partial x} + \frac{\tau_x^s - \tau_x^b}{\rho}$$

of the first momentum equation, where the pressure gradient term is written directly with respect to the water surface elevation  $\eta$ . This form of the momentum equations is given for example in Balzano's review paper [11].

In the *non-conservative formulation* of the SW equations, the pressure gradient term is based on the water surface elevation as well. The first momentum equation is rewritten as

$$\frac{\partial v_1}{\partial t} + v_1\frac{\partial v_1}{\partial x} + v_2\frac{\partial v_1}{\partial y} - fv_2 = -g\frac{\partial \eta}{\partial x} + \frac{\tau_x^s - \tau_x^b}{H\rho}$$

and the second momentum equation is rewritten accordingly. Therefore, numerical schemes based on the non-conservative formulation generally behave well for lake at rest steady states. However, in dry areas, the non-conservative form does not admit reasonable values for the velocity since this quantity is in fact not defined in dry areas and artificially setting  $\mathbf{v}$  to zero would lead to a discontinuity at a moving wet/dry front. In addition, the non-conservative form does not hold across shocks or hydraulic jumps.

To alleviate the well-balancedness issue, some methods also use the so called *pre-balanced* SW equations. These equations were designed to directly account for the balance of pressure forces acting on a fluid control volume and employ the water surface elevation  $\eta$  as a prognostic variable instead of the water height  $H$ . According to Liang and Marche in [114], the main advantage of the pre-balanced formulation is that it maintains the hyperbolicity of the original, conservative formulation and mathematically balances the flux and source terms at the same time. More precisely, the sum of pressure terms  $\frac{g}{2}\frac{\partial H^2}{\partial x} + gH\frac{\partial}{\partial x}b$  is rewritten in the variables of surface elevation and bottom topography as

$$\frac{g}{2}\frac{\partial H^2}{\partial x} + gH\frac{\partial}{\partial x}b = \frac{g}{2}\frac{\partial(\eta - b)^2}{\partial x} + g(\eta - b)\frac{\partial}{\partial x}b$$

and algebraically manipulated to obtain the form

$$\frac{g}{2}\frac{\partial(\eta^2 - 2\eta b)}{\partial x} + g\eta\frac{\partial}{\partial x}b.$$

If  $\eta = \text{const}$ , both of the above summands have precisely the same form  $g\eta\frac{\partial}{\partial x}b$ . The pre-

balanced form of the SW equations is now given by the equations

$$\begin{aligned} \frac{\partial \eta}{\partial t} + \frac{\partial Hv_1}{\partial x} + \frac{\partial Hv_2}{\partial y} &= 0, \\ \frac{\partial Hv_1}{\partial t} + \frac{\partial \left( \frac{(Hv_1)^2}{\eta-b} + \frac{g}{2}(\eta^2 - 2\eta b) \right)}{\partial x} + \frac{\partial \frac{Hv_1 Hv_2}{\eta-b}}{\partial y} - f Hv_2 &= -g\eta \frac{\partial b}{\partial x} + \frac{\tau_x^s - \tau_x^b}{\rho}, \\ \frac{\partial Hv_2}{\partial t} + \frac{\partial \frac{Hv_1 Hv_2}{\eta-b}}{\partial x} + \frac{\partial \left( \frac{(Hv_2)^2}{\eta-b} + \frac{g}{2}(\eta^2 - 2\eta b) \right)}{\partial y} + f Hv_1 &= -g\eta \frac{\partial b}{\partial y} + \frac{\tau_y^s - \tau_y^b}{\rho}. \end{aligned}$$

Last but not least, we recall the skew-symmetric momentum formulation (1.167) introduced in Section 1.5.1 an analog of which can also be derived for the two-dimensional SW equations, thereby facilitating the construction of well-balanced numerical schemes as proposed in [57].

## 4.2 Numerical challenges of shallow water flow simulation

As already indicated at the beginning of this chapter, from the numerical standpoint, several challenges in dealing with wetting and drying in a shallow water model have been stated in the literature.

First, a numerical scheme simulating wetting and drying must be positivity preserving, i.e. the water depth must remain non-negative in the entire computational domain at all times since the governing equations are ill-posed for  $H < 0$  and the surface wave celerity  $\sqrt{gH}$  is not defined in this case. In addition, the computation of velocity from the discharge  $\mathbf{v} = (H\mathbf{v})/H$  becomes ill-posed as  $H$  approaches zero.

Second, the model must be locally and globally conservative. The SW equations are a system of balance laws which is generally formulated in the conserved quantities of water volume and momentum as in (4.1). Hereby, the volume conservation property may be seen as essential in most applications, especially in long-term environmental applications where even a small deviation may accumulate over long integration times. Conservation of momentum, on the other hand, is less crucial as the coastal ocean is generally highly dissipative. Nevertheless, a proper resolution of the momentum equation is required for correct representation of the advancing wet/dry front.

Third, reiterating the discussion in Section 1.5.1, the SW equations admit certain steady state solutions, most importantly the lake at rest situation which consists of a vanishing velocity vector and a constant sea surface elevation. In this situation the potentially non-zero flux gradients are exactly balanced by the non-zero source term due to non-flat bottom topography. A numerical scheme which does not discretely preserve this particular steady state is prone to instabilities as it may generate unphysical oscillations due to the improper balance of flux and source terms which may also affect the simulation of wetting and drying processes. In addition, such a scheme will experience difficulties to achieve lake at rest steady state solutions in the long time limit. Therefore, much effort has been taken to construct well-balanced schemes preserving the lake at rest steady state in a discrete sense, sometimes by reformulating the SW equations in terms of the surface elevation instead of the water height. It should be remarked in this context that it is still not trivial to discretely preserve lake at rest steady states in partially dry cells, e.g. at a shoreline, see for instance [22, 81].

A more specific requirement is the non-permeability of dry areas. Coastal domains often feature lakes or ponds that remain wet in the dry stage and become disconnected from larger water bodies. Such emerging dry barriers should remain impermeable, i.e. attain a zero volume flux so that the lakes do not artificially dry out. This condition is often violated by so called porosity schemes that relax the positivity requirement and/or allow water flow beneath the bed.

As water depth reduces, the flow is mostly dominated by the pressure gradient and bottom friction terms. Over a sloping bathymetry the pressure gradient remains non-zero until the water is completely drained, whereas the bottom friction parameterization increases slowing down the flow. The bottom friction term is problematic as it grows without bound as the depth vanishes. The non-zero pressure gradient, on the other hand, becomes an issue in methods that retain a thin water layer over the dry areas. Therefore, in order to ensure positivity, the *artificial pressure gradient* must be omitted or canceled. In the literature several methods have been developed for achieving this goal, including flux-limiting schemes, direct cancellation of the pressure gradient term, and positivity preserving limiters.

In addition to the properties listed above, an ideal numerical wetting and drying method should also be robust, computationally efficient, and generalizable to unstructured meshes. Robustness implies that the scheme remains stable under rapid flows and highly variable bathymetry. In terms of computational efficiency, many wetting and drying schemes introduce time step limitations which may increase the computational cost significantly and restrict their applicability to realistic problems.

Many techniques to deal with wetting and drying have been suggested and have been classified in at least two classes of methods – *moving mesh methods* and *fixed mesh methods*. With *fixed mesh methods*, the computational grid itself is fixed throughout the computation of the time-dependent solution. On the other hand, *moving mesh methods*, also referred to as mesh adaption algorithms adapt the boundaries of the computational mesh to precisely match the water front. Hence, many difficulties can be circumvented as the equations are always well-defined and there is no artificial pressure gradient. Thus, shorelines can be tracked quite accurately and a non-negative water height is present throughout the computational domain. However, as discussed in [141, 28, 107], moving mesh methods are computationally more expensive than fixed grid techniques, more difficult to implement especially in case of strongly varying bathymetry and complex boundary shapes. They can potentially lead to excessively elongated elements along the coastline as stated in [141], and according to [28], mesh adaption techniques do not necessarily yield more accurate solutions than fixed grid schemes. In addition, front-tracking is difficult to combine with implicit time integration.

Considering methods on fixed computational grids, a contemporary review of wetting and drying algorithms is given by Medeiros and Hagen [130]. It classifies the wetting and drying fixed mesh methods into four general frameworks: (1) *thin film algorithms*, (2) *element removal methods* that employ checking routines to determine if an element or a node is wet, dry or potentially one of the two, subsequently adding or removing nodes from the computational domain (3) *fluid depth extrapolation* from wet nodes onto dry ones, (4) *negative water depth methods*. As stated in [130], the defining feature of *thin film algorithms* is the constant presence of a small layer of water within the domain. The algorithm may distinguish between wet and dry cells only by a minimum water height threshold. When the water height drops

below the threshold, the velocity of the flow is often set to zero and fluxes between adjacent dry cells are prohibited. Commonly, in finite volume and discontinuous Galerkin schemes, a flux-limiting strategy is employed, where the fluxes are modified, i.e. reduced or canceled, in the vicinity of dry zones. Recent thin film algorithms have been developed in [54, 28, 73, 214, 213, 209, 132, 197, 106]. As discussed in [28], since thin film approaches keep a small layer of water in nominally dry regions until these cells become fully wet again, it is difficult to determine an exact shoreline. In addition, an erroneous gradient may be present at the shoreline, possibly generating unphysical flows which are difficult to remove without violating momentum conservation. A recent example of a finite volume implementation of wetting and drying via a thin layer approach is given by Warner et al. in [209]. The minimum depth used within their algorithm is a spatially constant user-defined parameter. If the total depth at the cell center is smaller than this thin layer tolerance, the cell is considered dry and no water is permitted to leave the cell, however water is allowed to enter dry cells at any time. Thus, the arrival of incoming tide is not limited. In addition, the authors state that if the blocking of water relied on the water depth at cell faces, isolated wet patches could be generated for a fluid depth below the tolerance, since the fluid depth at the cell center could in fact be higher.

For *element removal algorithms*, wet elements are included in the computational domain while dry ones are not. At the wetting front, further consideration is needed for the treatment of partially wet cells. For instance, this may include distinguishing partially wet cells of dam-break type from those of flooding type as in [16]. As discussed in [28], mesh reduction techniques may cause oscillations due to sudden elimination and addition of nodes as well as mass and momentum loss. Considering our list of desirable numerical properties, mass conservation and well-balancedness are hence the most endangered properties when an element removal algorithm is applied.

*Depth extrapolation methods* focus on the advancing water front from which information is extracted. Mostly, the fluid depth is extrapolated from wet cells onto dry ones, if the algorithm detects an advancement of the front. If new wet cells occur, the corresponding velocities are calculated. In this category, only few approaches are listed by Medeiros and Hagen and it is mentioned that these schemes occasionally lead to artificially wetted elements. In addition, mass conservation has to be dealt with by correction routines.

*Negative depth algorithms* allow the water surface to drop below the bottom topography, similar to the idea of porosity schemes. While regions with negative depth are considered as dry, fluid flow below the ground is dealt with a porosity approach. The concept of artificial porosity is based on assuming a certain porosity of the sea bed which has to be properly modeled, e.g. by a thin porous layer, and allows for non-zero fluxes in regions formerly considered as dry. The negative depth algorithms are the most recent schemes listed by Medeiros and Hagen. The benefit of negative depth methods is that there is no need to detect dry elements or cancel fluxes. In these methods the continuous equations are modified slightly to account for the porosity. As such the artificial pressure gradient issue does not arise, and these schemes are compatible with many discretizations, including implicit time integration schemes. However, negative depth algorithms often break the non-permeability requirement mentioned above. An alternative approach to negative depth algorithms has been used in [96] where the bottom topography is allowed to move in time as the water surface drops.

In the literature, explicit time stepping is implemented in the majority of previous wetting and

drying methods. In addition to being easier to implement, explicit time integration schemes are usually robust and provide physically sound results for wetting and drying as they can accurately represent the dynamics of the flooding front due to the necessarily small time step sizes. Moreover, explicit time integration is required by most flux-limiting methods as well as thin layer methods detecting dry elements.

However, explicit schemes are subject to time step limitations that may become severe, especially if the mesh is refined in the shallow regions. Therefore, multirate explicit time stepping using local time steps depending on the location of the flow variables within the computational domain has been applied to geophysical flows e.g. by Seny et al in [174]. The choice of explicit or implicit time integration should also be based on the involved time scales of the specific application. In an evaluation of several methods which were developed until 1994 for the simulation of wetting and drying in one space dimension, Balzano [11] already includes several implicit schemes. In addition, a simple calculation is given regarding the distance which the moving boundary covers within one time step. More precisely, if the speed of the moving boundary is  $v_b$  and  $\Delta t$  denotes the time step, we have for this distance

$$v_b \Delta t = \frac{v_b}{\sqrt{gH}} \frac{\Delta t}{\Delta x} \sqrt{gH} = \text{Fr Cr} \Delta x,$$

where  $\text{Fr}$  and  $\text{Cr}$  are the Froude number and the Courant number, respectively,  $\Delta x$  is the cell size,  $H$  the water height and  $g$  is the gravitational constant. Balzano further argues that, in practice, the Courant number is between 0.5 and 1 while the Froude number takes typical values in the interval  $[0.01, 0.05]$ , hence 20 to 200 time steps would be needed to move the wet/dry boundary over a distance of  $\Delta x$ , i.e. a cell length.

Considering the possibly severe limitation of the allowable time step size, devising implicit time stepping schemes compatible with wetting and drying is generally desirable. In fact, the interest in the further development of implicit schemes for this type of application has increased in recent years.

A special unconditionally positive implicit time integration scheme is used by Casulli [38]. This approach leads to a mildly nonlinear system to be solved each time step but is mass conserving and guarantees non-negative water height for any time step size. However, this method is only first order accurate in space and time. In the context of stabilized residual distribution schemes, Ricchiuto and Bollermann [168] developed a well-balanced and positivity preserving scheme for shallow water flows considering implicit time integration via the second order trapezoidal rule. In this case, the time step size can be chosen twice as large as for the explicit Euler scheme. A different approach is taken by Kärnä et al. [96]. There, the bottom topography is allowed to move in time as water elevation drops, i.e. a user-defined function is introduced which redefines the bottom topography. However, this function has to fulfill certain conditions and hence has to be carefully chosen prior to numerical computation. In [132], Meister and Ortlob developed an unconditionally positive approach to time integration within a DG scheme in order to obtain non-negative water heights without time step restriction. This approach is based on the MPSDIRK3 modification of the classical third order SDIRK scheme by Cash as described in Section 3.4.3 and will be discussed in more detail in Section 4.4. Further recent implicit approaches are given by Marras et al. [125], combining dynamic viscosity for shock capturing with a high-order wetting and drying method, a 3D non-hydrostatic implicit model by Candy [31], a finite difference implicit 1D code by Kalita

and Sarma [94], and a thin layer DG algorithm with implicit time integration developed by Le et al. [106] adopting a regulation of gravitational forces by a blending parameter in partially dry cells to enable fast Newton convergence of the implicit solver.

### 4.3 Review of wetting and drying procedures for FV and DG methods

In recent decades, finite volume (FV) ocean circulation models have become increasingly popular compared to previously used methods based on finite differences which is due to the finite volume conservation properties, their suitability for advection dominated problems, and their applicability on quite general grids. In addition, there has been an increasing interest in developing higher-order methods since they require fewer degrees of freedom to obtain the desired accuracy. Particularly the discontinuous Galerkin (DG) schemes are attractive due to their support of arbitrary meshes, their amenability for hp-refinement and parallel computation, in addition to their favorable dissipation and dispersion properties and their similarity to finite volume schemes in terms of local conservation properties. As expounded in this section, the wetting and drying treatments used within FV or DG schemes differ in basic construction principles.

#### 4.3.1 Finite Volume Methods

Finite volume methods for the SW equations commonly start from the conservative formulation based on either the water height and discharge or the surface elevation and discharge. While the numerical representation of the conserved quantities in a finite volume scheme is given by their cell averages, the bottom topography is typically represented either by cell means as in [10, 124, 114, 52, 41] or by a piecewise linear function which is continuous across cell boundaries as in [105, 27, 81]. For the one-dimensional SW equations only including the pressure forces due to non-flat bottom topography and neglecting bottom friction as well as Coriolis forces, the governing equations are given by

$$\frac{\partial}{\partial t} \mathbf{U}(x, t) + \frac{\partial}{\partial x} \mathbf{F}(\mathbf{U}(x, t)) = \mathbf{S}(\mathbf{U}(x, t), x), \quad (4.5)$$

with  $\mathbf{U}(x, t) = (H, Hv)^T$ ,  $\mathbf{F}(\mathbf{U}) = (Hv, Hv^2 + \frac{g}{2}H^2)^T$ ,  $\mathbf{S} = (0, -gH\frac{\partial b}{\partial x})$ . The basic FV scheme discretizing the above equations in space has the form

$$\Delta x_i \frac{d}{dt} \mathbf{U}_i(t) + \mathbf{F}_{i+1/2}^* - \mathbf{F}_{i-1/2}^* = \mathbf{S}_i, \quad \mathbf{U}_i = \begin{pmatrix} H_i \\ H_i v_i \end{pmatrix}, \quad (4.6)$$

where  $\mathbf{U}_i$  contains the cell means, the quantities  $\mathbf{F}_{i+1/2}^* = \mathbf{F}^*(\mathbf{U}_i, \mathbf{U}_{i+1})$  and  $\mathbf{F}_{i-1/2}^* = \mathbf{F}^*(\mathbf{U}_{i-1}, \mathbf{U}_i)$  denote the output of a suitable numerical flux function  $\mathbf{F}^*$  and  $\mathbf{S}_i$  is a suitable source term discretization.

#### Achieving well-balancedness

As stated in Section 4.1, due to the conservative formulation, special consideration has to be taken to ensure the well-balanced property. More precisely, the necessary cancellation of

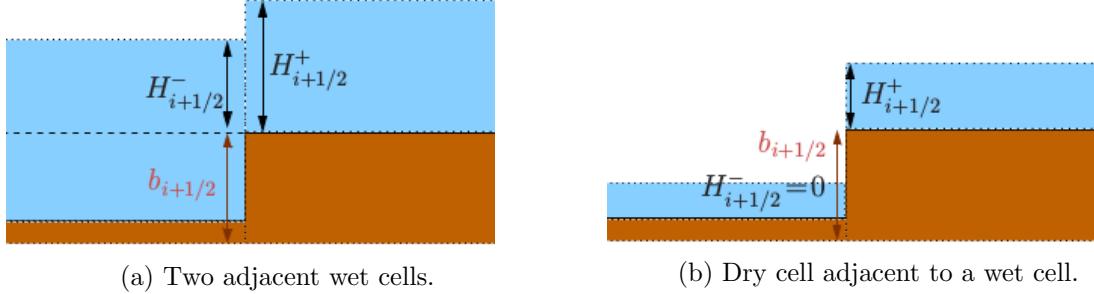


Figure 4.1: Illustration of hydrostatic reconstruction approach depicting auxiliary interface bottom topography  $b_{i+1/2}$  and reconstructed left and right water heights  $H_{i+1/2}^\pm$ .

pressure terms in equation (4.4) for the lake at rest is not fulfilled by the simple source term discretization

$$-gH \frac{\partial}{\partial x} b \approx -gH_i \frac{b_{i+1} - b_{i-1}}{\Delta x}$$

coupled with a classical numerical flux such as the Lax-Friedrichs (LF) flux or the Harten-Lax-van Leer (HLL) flux [76]. This fact has also been revisited by Clain et al. in [41].

Therefore, in [10], Audusse et al. suggested a very simple technique handling both discontinuous topography and wet/dry interfaces which has become quite popular by now. Their method will be reviewed in the following, using the illustration in Figure 4.1. In order to determine the input values for the numerical flux function at a cell interface, the water surface  $\eta = H + b$  and the bottom topography on the two adjacent cells are taken into account. In a first order scheme, at an interface denoted by the index  $i + 1/2$ , the reconstructed left and right water height is given by the non-negative values of

$$\begin{aligned} H_{i+1/2}^- &= \max\{0, H_i + b_i - b_{i+1/2}\} \\ H_{i+1/2}^+ &= \max\{0, H_{i+1} + b_{i+1} - b_{i+1/2}\}, \end{aligned} \tag{4.7}$$

where  $b_{i+1/2} = \max\{b_i, b_{i+1}\}$ .

The clipped water height values  $H_{i+1/2}^\pm$  now determine the left and right states of the conserved variables. They are computed as  $\mathbf{U}_{i+1/2}^\pm = \begin{pmatrix} H_{i+1/2}^\pm \\ H_{i+1/2}^\pm v_{i+1/2}^\pm \end{pmatrix}$ , where  $v_{i+1/2}^- = v_i$  and  $v_{i+1/2}^+ = v_{i+1}$ .

These values are then used as input values for the numerical flux function in (4.6), i.e. the basic FV scheme is modified by inserting

$$\mathbf{F}_{i\pm 1/2}^* = \mathbf{F}^*(\mathbf{U}_{i\pm 1/2}^-, \mathbf{U}_{i\pm 1/2}^+) \tag{4.8}$$

for the numerical flux values. In order to achieve well-balancedness, the source term discretization is based on the reconstructed values of water height. If the source term simply contains the pressure forces given by the term  $-gHb_x$  due to the bottom slope and all other forces such as bottom friction and Coriolis forces are neglected, its discretization is given by

$$\mathbf{S}_i = \frac{g}{2} \begin{pmatrix} 0 \\ (H_{i+1/2}^-)^2 - (H_{i-1/2}^+)^2 \end{pmatrix}.$$

As mentioned in [10], this source term discretization can now be redistributed to the cell interfaces. The scheme (4.6) using hydrostatic reconstruction via the numerical fluxes (4.8) can then be rewritten as

$$\Delta x_i \frac{d}{dt} \mathbf{U}_i(t) + \mathbf{F}_l^*(\mathbf{U}_i, \mathbf{U}_{i+1}, b_i, b_{i+1}) - \mathbf{F}_r^*(\mathbf{U}_{i-1}, \mathbf{U}_i, b_{i-1}, b_i) = 0, \quad (4.9)$$

where the left and right interface fluxes are given by

$$\begin{aligned} \mathbf{F}_l^*(\mathbf{U}_i, \mathbf{U}_{i+1}, b_i, b_{i+1}) &= \mathbf{F}^*(\mathbf{U}_{i+1/2}^-, \mathbf{U}_{i+1/2}^+) + \left( \begin{array}{c} 0 \\ H_i^2 - (H_{i+1/2}^-)^2 \end{array} \right), \\ \mathbf{F}_r^*(\mathbf{U}_i, \mathbf{U}_{i+1}, b_i, b_{i+1}) &= \mathbf{F}^*(\mathbf{U}_{i+1/2}^-, \mathbf{U}_{i+1/2}^+) + \left( \begin{array}{c} 0 \\ H_{i+1}^2 - (H_{i+1/2}^+)^2 \end{array} \right). \end{aligned} \quad (4.10)$$

In order to achieve a spatial discretization of second order, a piecewise linear representation of the conserved quantities needs to be reconstructed in a way which maintains well-balancedness and non-negativity. Thus, starting with a first order FV scheme, a second order extension may be obtained using reconstructed values at the cell interface. Thereby, gradients of the conserved quantities are computed and limited if necessary, in order to avoid overshoots. Based on the situation at an interface between a wet and a dry cell, Audusse et al. argue that of the quantities  $H, b$  and  $\eta = H + b$ , the water height  $H$  and water surface  $\eta$  are the variables which should actually be reconstructed while  $b$  is to be computed as  $b = \eta - H$ .

### Positivity preservation

In case of wetting and drying, one of the basic ingredients to ensure non-negativity of water height within a FV discretization is a positivity preserving numerical flux. Used within a FV method and explicit Euler time integration, these numerical flux functions yield non-negative cell means of water height at the next time level under the premise of non-negative cell means at the current time level. Roe-type solvers based on the linearized original equations and the corresponding modified Riemann problems are generally not positivity preserving which is discussed in detail by Pelanti et al. in [156] for the specific case of the SW equations. As a remedy for the Roe solver, positivity in [156] is guaranteed by a relaxation solver modifying Roe's method. Several other positivity preserving fluxes may be used, e.g. the classical or local Lax-Friedrichs flux or the HLL flux. If the well-balanced flux (4.10) designed by Audusse et al. is based on the Lax-Friedrichs flux for  $F^*$ , positivity in one space dimension can be shown as follows, see also [214]. Hereby, we consider the first order finite volume scheme (4.9) using the interface fluxes (4.10) where  $\mathbf{F}^*$  is the Lax-Friedrichs flux

$$\mathbf{F}^*(\mathbf{U}_l, \mathbf{U}_r) = \frac{1}{2} (\mathbf{F}(\mathbf{U}_l) + \mathbf{F}(\mathbf{U}_r) - \alpha(\mathbf{U}_r - \mathbf{U}_l)), \quad \alpha = \max \left\{ |v| + \sqrt{gH} \right\},$$

where the maximum is taken over the cell means in the whole region. Using explicit Euler time integration, the cell averages of water height  $H_i^n, H_i^{n+1}$  at two successive time levels  $t^n, t^{n+1}$  are then related by

$$H_i^{n+1} = H_i^n - \frac{\Delta t}{\Delta x_i} \left( F_1^*(\mathbf{U}_{i+1/2}^-, \mathbf{U}_{i+1/2}^+) - F_1^*(\mathbf{U}_{i-1/2}^-, \mathbf{U}_{i-1/2}^+) \right), \quad (4.11)$$

where  $F_1^*$  is the first component of the numerical flux.

**Lemma 4.1.** *Let all cell averages  $H_i^n$  of the water height at time  $t^n$  be non-negative and let the cell averages  $H_i^{n+1}$  at time  $t^{n+1}$  be computed by the finite volume scheme (4.11) such that the time step  $\Delta t$  fulfills the CFL condition  $\alpha \frac{\Delta t}{\Delta x_i} \leq 1$ . Then the cell averages  $H_i^{n+1}$  are non-negative as well.*

*Proof.* First, we note that the first component of the Lax-Friedrichs numerical flux is given by

$$F_1^*(\mathbf{U}_l, \mathbf{U}_r) = \frac{1}{2}((Hv)_l + (Hv)_r - \alpha(H_r - H_l)).$$

Now, since  $\mathbf{U}_{i+1/2}^- = \frac{H_{i+1/2}^-}{H_i}(H_i, (Hv)_i)^T$  and  $\mathbf{U}_{i+1/2}^+ = \frac{H_{i+1/2}^+}{H_{i+1}}(H_{i+1}, (Hv)_{i+1})^T$ , we can rewrite the new water height average as

$$\begin{aligned} H_i^{n+1} &= \left( 1 - \frac{\Delta t}{2\Delta x_i} \frac{H_{i+1/2}^{n,-}}{H_i^n} (v_i^n + \alpha) - \frac{\Delta t}{2\Delta x_i} \frac{H_{i-1/2}^{n,+}}{H_i^n} (\alpha - v_i^n) \right) H_i^n \\ &\quad + \frac{\Delta t}{2\Delta x_i} \frac{H_{i-1/2}^{n,-}}{H_{i-1}^n} (\alpha + v_{i-1}^n) H_{i-1}^n + \frac{\Delta t}{2\Delta x_i} \frac{H_{i+1/2}^{n,+}}{H_{i+1}^n} (\alpha - v_{i+1}^n) H_{i+1}^n. \end{aligned}$$

Per construction in (4.7) we have  $H_{i+1/2}^- \leq H_i$  and  $H_{i-1/2}^+ \leq H_i$ . Thus, the factor in front of  $H_i^n$  rewrites as

$$1 - \frac{\Delta t}{2\Delta x_i} \frac{H_{i+1/2}^{n,-}}{H_i^n} (v_i^n + \alpha) - \frac{\Delta t}{2\Delta x_i} \frac{H_{i-1/2}^{n,+}}{H_i^n} (\alpha - v_i^n) \geq 1 - \alpha \frac{\Delta t}{\Delta x_i} \frac{\max\{H_{i+1/2}^{n,-}, H_{i-1/2}^{n,+}\}}{H_i^n} \geq 0$$

due to the CFL condition  $\alpha \frac{\Delta t}{\Delta x_i} \leq 1$ . Furthermore, as  $\alpha \geq \max\{|v_i^n|, |v_{i+1}^n|\}$ , the factors in front of  $H_{i-1}^n$  and  $H_{i+1}^n$  are non-negative as well. This proves non-negativity of the cell average  $H_i^{n+1}$ .  $\square$

The well-balanced and positivity preserving approach by Audusse et al. is simple and fulfills the requirements of well-balancedness and non-negativity. It has thus been used as a basic building block in many subsequent FV schemes in one space dimension such as [124, 114] as well as extension to unstructured meshes in [52] and structured meshes in [41]. It should be remarked that both of the approaches by Liang and Marche in [114] and by Duran et al in [52] are based on the pre-balanced formulation given in Section 4.1. Using a positivity preserving numerical flux for the conservative form of the SW equations is thereby sufficient to guarantee non-negative cell means of water height also for the pre-balanced formulation. For the Lax-Friedrichs flux, this has been shown by Duran and Marche in [53], while for the HLL flux introduced in [76], positivity preservation for the pre-balanced formulation of the SW equations has been proven by Meister and Ortlob [131] for a space discretization by the DG scheme incorporating finite-volume subcells.

The possibility to rewrite the approach by hydrostatic reconstruction in terms of only interface fluxes which incorporate the source term as in (4.9) has led to its use as an important ingredient within many of the discontinuous Galerkin schemes to be reviewed in Section 4.3.2. In this respect, the approach of hydrostatic reconstruction simply yields a modification of the numerical flux function used within the discontinuous Galerkin scheme in case of a discontinuous representation of the bottom topography.

FV schemes based on a continuous, piecewise linear representation of the bottom topography need slightly different techniques to guarantee well-balancedness and non-negativity at the same time. Approaches of this kind are usually found within the class of central-upwind FV schemes as in [105, 27, 81]. The term central-upwind hereby refers to the use of a specific numerical flux function which is a weighted sum of a central and an upwind part with weights determined by the computed characteristic speeds. The central-upwind scheme by Kurganov and Petrova [105] can essentially be written in the form (4.6) where the conserved variables are now taken as the water surface and the discharge, i.e.  $\mathbf{U}_i = \begin{pmatrix} \eta_i \\ (Hv)_i \end{pmatrix}$  and the source term is computed from the linear bottom topography and the water surface as  $\mathbf{S}_i = \begin{pmatrix} 0 \\ -g(\eta_i - b_i)(b_{i+1/2} - b_{i-1/2}) \end{pmatrix}$ .

Kurganov and Petrova also explicitly mention the ill-conditioned computation of the velocity by  $v = \frac{Hv}{H}$  for very small water height which is needed to evaluate the flux function as well as the numerical fluxes and propose to avoid the division by small numbers via the formula

$$v := \frac{\sqrt{2}H(Hv)}{\sqrt{H^4 + \max\{H^4, \epsilon\}}}, \quad (4.12)$$

with the regularization parameter  $\epsilon$  chosen grid dependent, i.e. decreasing with decreasing grid size. The algorithm then recomputes the discharge  $Hv$  using  $(Hv) := H \cdot v$  where it is explicitly mentioned that failing to adjust the discharge may produce negative values of  $H$ , in accordance with the proof of non-negativity in [105]. Furthermore, the scheme by Kurganov and Petrova has been extended to triangular grids in [27].

Regarding implicit FV schemes, fewer wetting and drying methods are reported. One of the few algorithms which guarantee non-negative water height independent of the time step size is the first order FV method by Casulli [38]. It is based on the non-conservative formulation of the SW equations which is discretized in time in a semi-implicit way. The precise formulation is vital to obtain an M-matrix property for the matrices used within the iterations of the Newton-type scheme in order to achieve positivity. Moreover, using the properties of the Jacobians, it can furthermore be shown that the Newton iteration generates a converging sequence to the solution of the semi-implicit scheme.

### 4.3.2 Discontinuous Galerkin Schemes

The advancement of wetting and drying techniques for discontinuous Galerkin schemes is more recent than for finite volume methods. In this context, Bokhove [21] developed a space-time DG scheme which uses a mesh adaption strategy to accurately separate wet and dry regions. Regarding the method-of-lines framework, the thin layer approaches of Ern et al. [54] and Bunya et al. [28] generally provide the background for current DG approaches providing both a well-balanced discretization and the ability to deal with wetting and drying.

#### Achieving well-balancedness

In order to preserve lake at rest steady state solutions, the hydrostatic reconstruction technique may be transferred from finite volume schemes to DG schemes. According to the

discussion of the discontinuous Galerkin framework in Section 1.2, the classical derivation of a DG scheme consists in multiplying the governing equations by test functions and integrating over the computational domain. As the governing equations, we now consider the SW equations in compact conservative form (4.2) on the computational domain  $\Omega \times \mathbb{R}_+$ , with  $\mathbf{x} = (x, y)^T \in \Omega$  and  $t \in \mathbb{R}_+$ , where  $\Omega \subset \mathbb{R}^2$  is an open polygonal domain. Of course, initial conditions  $\mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x})$  and appropriate boundary conditions are assumed to be given. Unstructured triangular grids allow for a greater flexibility in discretizing spatial domains. As in Section 1.2.5, we will therefore consider a conforming triangulation  $\mathcal{T}^h$ , consisting of triangular elements  $\tau_i$  of the given computational domain  $\Omega$ . Let  $\mathcal{W}^h$  be the piecewise polynomial space defined by

$$\mathcal{W}^h = \{w_h \in L^\infty(\Omega) \mid w_h|_{\tau_i} \in \mathcal{P}^N(\tau_i) \quad \forall \tau_i \in \mathcal{T}^h\},$$

where  $\mathcal{P}^N(\tau_i)$  denotes the space of all polynomials on  $\tau_i$  of degree  $\leq N$ .

Now, a DG approximation to the exact solution of the conservative SW equations is given by a vector of piecewise polynomial functions  $\mathbf{U}_h(\cdot, t) \in (\mathcal{W}^h)^3$ , with

$$\mathbf{U}_h(\mathbf{x}, t) = (H_h(\mathbf{x}, t), (Hv_1)_h(\mathbf{x}, t), (Hv_2)_h(\mathbf{x}, t))^T,$$

which satisfies the semi-discrete equation

$$\begin{aligned} \frac{d}{dt} \int_{\tau_i} \mathbf{U}_h \cdot \mathbf{W} d\mathbf{x} &= \int_{\tau_i} \mathbf{F}_1(\mathbf{U}_h) \cdot \frac{\partial \mathbf{W}}{\partial x} + \mathbf{F}_2(\mathbf{U}_h) \cdot \frac{\partial \mathbf{W}}{\partial y} d\mathbf{x} - \int_{\partial \tau_i} \mathbf{F}^*(\mathbf{U}_i^-, \mathbf{U}_i^+, \mathbf{n}) \cdot \mathbf{W} d\sigma \\ &\quad + \int_{\tau_i} \mathbf{S}_h(\mathbf{U}_h, \mathbf{x}) \cdot \mathbf{W} d\mathbf{x} \end{aligned} \tag{4.13}$$

for any  $\tau_i \in \mathcal{T}^h$ ,  $\mathbf{W} \in (\mathcal{W}^h)^3$ , where the source term is discretized by

$$\mathbf{S}_h(\mathbf{U}_h, \mathbf{x}) = -g \cdot (0, H_h \cdot \partial_x b_h, H_h \cdot \partial_y b_h)^T,$$

with  $b_h$  being a suitable projection of the bottom topography  $b$  to  $\mathcal{W}^h$ . We recall that both the approximate DG solution and the projected bottom topography are allowed to be discontinuous across cell interfaces. Hence, the numerical flux function  $\mathbf{F}^*$  takes into account the left-hand and right-hand sided values  $\mathbf{U}_i^-$ ,  $\mathbf{U}_i^+$  of the approximate solution within  $\tau_i$  and an adjacent element, respectively, and is dependent on the outward pointing normal vector  $\mathbf{n}$  of the cell  $\tau_i$ . The DG scheme may generally use any numerical flux function developed within the context of FV schemes. However, similar to finite volume schemes, a modification is necessary to preserve still water stationary states in case of non-constant bottom topography. Hereby, Ern et al. design a well-balanced scheme by incorporating the hydrostatic reconstruction numerical flux by Audusse et al. [10] described in Section 4.3.1, i.e. the numerical flux  $\mathbf{F}^*$  in (4.13) is replaced by the well-balanced flux (4.10) evaluated for the hydrostatically reconstructed left and right states. Thus, we replace the expression  $\mathbf{F}^*(\mathbf{U}_i^-, \mathbf{U}_i^+, \mathbf{n})$  in (4.13) by  $\mathbf{F}^{WB}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, H_i^-, \mathbf{n})$ , where the well-balanced numerical flux denoted by  $\mathbf{F}^{WB}$  is adjusted to the case of unstructured triangular grids, i.e. we set

$$\mathbf{F}^{WB}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, H_i^-, \mathbf{n}) = \mathbf{F}^{num}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, \mathbf{n}) + \begin{pmatrix} 0 \\ \frac{g}{2} \left( (H_i^-)^2 - (H_{i,*}^-)^2 \right) n_1 \\ \frac{g}{2} \left( (H_i^-)^2 - (H_{i,*}^-)^2 \right) n_2 \end{pmatrix}, \tag{4.14}$$

where the modified (starred) left and right states are obtained from the hydrostatic reconstruction

$$\begin{aligned}\mathbf{U}_{i,*}^\pm &= (H_{i,*}^\pm, H_{i,*}^\pm \cdot (v_1)_i^\pm, H_{i,*}^\pm \cdot (v_2)_i^\pm)^T, \\ H_{i,*}^\pm &= \max \{0, H_i^\pm + b_i^\pm - \max \{b_i^-, b_i^+\}\}.\end{aligned}$$

Commonly, the DG scheme employs quadrature formulae to approximately solve the integrals involved in its variational formulation. However, although we obtain a local truncation error of order  $N + 1$  if the integrals in (4.13) are approximated by quadrature formulae of order  $2N$  on elements and of order  $2N + 1$  on edges, the well-balanced property in general requires a higher degree of exactness. More precisely, quadrature rules of order  $3N - 1$  over elements and of order  $3N$  over edges need to be implemented in the well-balanced DG scheme, see e.g. [133].

### Specifics of thin-layer wetting and drying approaches

In order to deal with wetting and drying and thus with vanishing water height or possibly negative values, Ern et al. [54] furthermore introduce a slope modification. Hereby, if the cell average of the water height is negative, the water height in this cell is set to zero. If the cell average of the water height  $H$  is non-negative, but the minimum over the integration points is below the drying threshold, a linear representation of  $H$  is reconstructed where the slope is modified. In addition the corresponding discharge is set to zero. As long as no negative averages of  $H$  occur, mass is preserved by this approach but not momentum. In addition, the method may add mass if non-negativity is violated. In order to rectify the artificial increase of mass, Bunya et al. redistribute the water mass within an element by a modification of the surface elevation to guarantee non-negativity. Additionally, they thereby guarantee local mass conservation. The redistribution process heavily relies on non-negative averages of the water height, so both a sufficient condition on the allowable time step to guarantee non-negativity is derived and a reflection flux is introduced to dispose of the time step restriction. Furthermore, fluxes between dry cells are restricted to prevent unphysical oscillations and to prevent dry cells from loosing their mass.

As stated in Kärnä et al. [96], the main difficulty in thin layer approaches when used within higher order methods such as DG is treating the wet/dry transition elements. These cells may have “hanging nodes” above the water surface elevation possibly causing an artificial pressure gradient, i.e. gravity forces that drag the water down, as shown in Figure 4.2. Such a situation may also occur for the lake at rest because, in general, the prolongation of the constant water surface by the dry bottom topography is not a smooth function. In fact, its approximation by a polynomial function then usually yields hanging nodes. This can be seen in Figure 4.3a where obviously the water surface elevation has an artificial slope creating an artificial pressure gradient. This artifact does not occur for moving mesh methods adapted to the shore line as depicted in Figure 4.3b and is also prevented in negative depth algorithms. In this context, in order to address this problem in a numerical method, a distinction between dam-break type (Figure 4.3c) and flooding-type partially dry cells (Figure 4.3a) has already been proposed by Bates and Hervouet in [16]. In case of flooding-type cells which by definition also include lake at rest situations, movement of water in dry cells is only allowed by convective

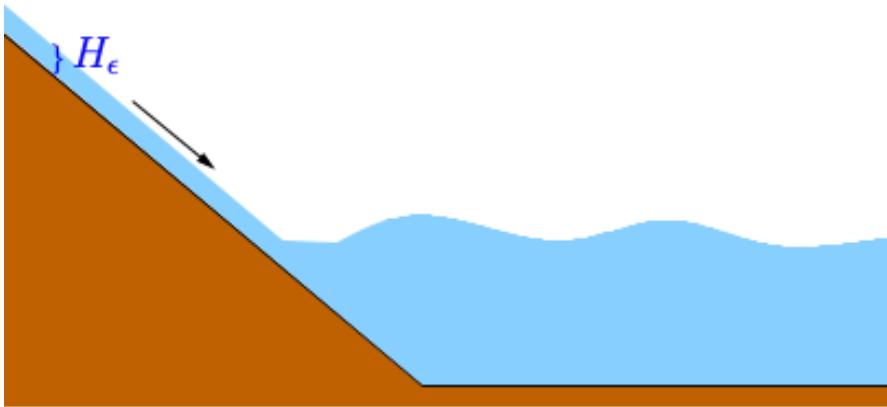


Figure 4.2: Illustration of the numerical solution of a typical thin layer method. We note that the method conserves a small layer of water of height  $H_\epsilon$  in an actually dry region, hence artificial gravity forces are present due to hanging nodes above the water surface.

transport. However, for dam-break cells a high water surface level next to the dry cell results in the flow of water due to gravitational forces, which should not be neglected in this case. Flooding-type cells can be identified by considering the representation of water surface and bottom topography within the numerical scheme. Given a cell  $\tau_i$  with water surface elevation  $\eta_h(x, y)$  and bottom representation  $b_h(x, y)$ , then  $\tau_i$  is a flooding-type partially dry cell if we have

$$\max_{(x,y) \in \tau_i} \eta_h(x, y) - \max_{(x,y) \in \tau_i} b_h(x, y) < H_\epsilon, \quad (4.15)$$

where  $H_\epsilon$  denotes the drying threshold under which a node is considered dry. This distinction of partially dry cells has also been made by other authors, for example by Bunya et al. [28], where it is included in the wet/dry status to determine fluxes between dry elements and by Vater et al. [197] in order to cancel gravitational forces from the semi-discretization in flooding type cells.

Once, partially dry cells are distinguished with respect to their desired behavior, the numerical methods are adjusted. Common to many high order schemes, Bunya et al. use cancellation of gravity to balance the effect of an artificial gradient of the surface elevation in wetting or drying elements. However, their precise choice of the momentum fluxes violates momentum conservation.

In addition, we have to remark that these more or less complicated rules which are applied to partially dry cells in order to remove the artificial pressure forces are difficult to integrate into an implicit time stepping scheme. For instance, in [73], a flux limiting wetting and drying DG approach is developed and applied to achieve a realistic simulation of the Scheldt Estuary. The limitation involves computing three intermediate states of the water surface elevation preserving local mass conservation. As in the method of Bunya et al. [28], gravitational forces are neglected within dry elements to allow the water surface to align with the bottom topography. Within a “buffer layer” of very shallow water, bottom stress and eddy viscosity are increased while surface stress is decreased. Due to the many switches in turning the fluxes on or off, the method is discontinuous with respect to the variables. Therefore, Gourge et al.

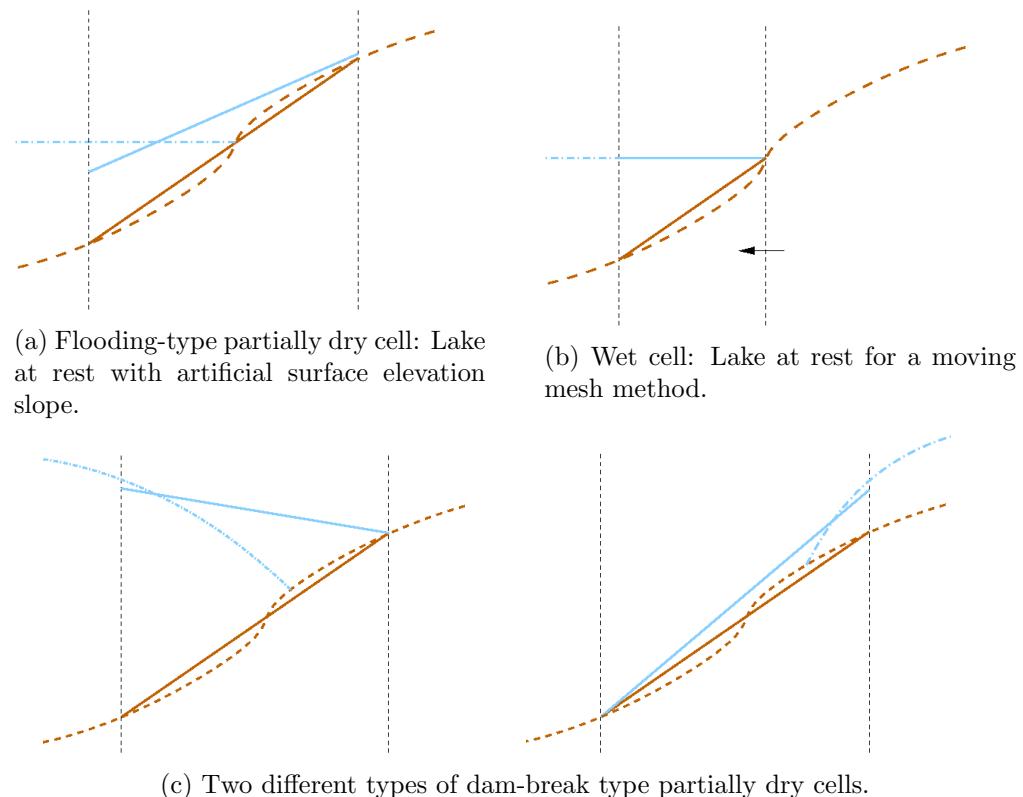


Figure 4.3: Illustration of surface elevation, bottom topography and shoreline representation in one discretization cell for a second order method in 1D. Black dashed vertical lines: cell boundaries. Blue and brown dashed lines: continuous surface elevation and bottom topography. Solid lines: discrete representation.

state that implicit time stepping is not directly available for this approach. By preventing the switches caused from the removal of artificial pressure forces, Le et al. [106] recently accomplished to combine the above distinction between dam break and flooding type cells with implicit time integration. Hereby, the condition (4.15) is incorporated into the calculation of a blending parameter in the interval  $[0, 1]$ . Instead of employing a basic on/off switch, the gravitational fluxes in the DG semi-discretization are then simply multiplied by this blending parameter, thereby facilitating fast convergence of the non-linear solver.

Additional difficulties may arise through the combination of the above wetting and drying techniques with TVB limiters for shock capturing. As reported by Ern et al. [54] and Bunya et al. [28], the slope modification of the wet/dry treatment and the TVB slope limiter of the DG scheme may artificially activate each other, possibly leading to instability. Thus, the TVB limiter is commonly only applied to the fully wet region.

The numerically determined velocity in nearly dry regions can be large, especially within a DG scheme which uses a polynomial representation of the conserved variables. In this context, the challenge of computing a stable linear distribution of the velocity within a DG scheme is addressed by Vater et al. in [197] via a velocity-based limiting procedure. Furthermore, favorable properties of monotonicity preserving Bernstein polynomials have been used within a DG model for flooding and drying by Beisiegel and Behrens in [18].

### A systematic approach to positivity preservation and well-balancedness

The series of works by Xing, Zhang and Shu with the basic ideas given in [214, 213] deal with the construction of a so-called positivity preserving limiter and the special design of explicit Runge-Kutta time integrators based on convex combinations of explicit Euler steps. Here, the methods are generally in line with the ideas in [54, 28], but a more systematic approach to well-balancedness and positivity preservation is given. Therefore, in principle, the properties of well-balancedness and positivity preservation for the SW equations carry over to high order in space and time. One of the basic ingredients of the method by Xing et al. [214] is a positivity preserving numerical flux function for the SW equations as introduced in Section 4.3.1. For these flux functions which attain non-negative cell means for the finite volume method with explicit Euler time stepping, positivity preservation can be extended both to a DG spatial discretization and to higher order Runge-Kutta schemes in time.

For the DG scheme complemented by explicit Euler time integration, let  $\mathbf{U}_h^n(\mathbf{x})$  and  $\mathbf{U}_h^{n+1}(\mathbf{x})$  denote the approximate solution at the time levels  $t^n$  and  $t^{n+1} = t^n + \Delta t$ , respectively, and let the cell averages at time  $t^n$  be denoted by  $\bar{\mathbf{U}}_i^n = \frac{1}{|\tau_i|} \int_{\tau_i} \mathbf{U}_h^n(\mathbf{x}) d\mathbf{x}$ . In order to carry over positivity preservation from the FV scheme to the DG scheme, the semi-discrete DG equations referring to the water height cell averages are rewritten as convex combinations of finite volume approximations. The occurring factors in this convex combination then determine the CFL condition for positivity preservation, see [214]. Hereby, the representation of the DG scheme is based on values at a set  $X_i^N \subset \tau_i$  of intermediate nodes within the DG cell, depending on the polynomial degree of the DG approximation. Using a corresponding distribution on the reference element, the set  $X_i^N$  related to the finite volume representation may be computed a priori. The result by Xing et al. [214, 213] is then summarized as follows.

**Theorem 4.2.** *For the DG scheme with positivity preserving numerical flux function and*

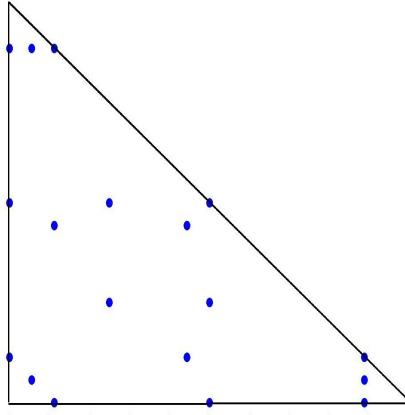


Figure 4.4: Nodes for positivity enforcement on triangular grids for a polynomial degree of  $N = 2$ .

explicit Euler time integration, the new cell averages  $\bar{H}_i^{n+1}$  of the water height at time  $t^{n+1}$  are non-negative under the premises that the time step  $\Delta t$  fulfills a suitable CFL-type restriction and that  $H_h^n(\mathbf{x})$  is non-negative at each node in the set  $X_i^N \subset \tau_i$ .

**Remark 4.3.** The construction of the nodal set  $X_i^N$  is described by Zhang et al. in [220]. Hereby, the nodes in  $X_i^N$  result from quadrature rules on the triangle  $\tau_i$  with positive weights that exactly integrate polynomials of degree  $N$ . Furthermore,  $X_i^N$  is supposed to contain the points that are used for numerical integration over the element boundaries  $\partial\tau_i$  within the DG scheme. To provide an example, on the reference triangle  $\mathbb{T} = \{(r, s) \in \mathbb{R}^2 \mid -1 \leq r, s, r + s \leq 0\}$  also considered in Section 1.2.5, the distribution of these nodes for  $N = 2$  is shown in Figure 4.4.

To enforce non-negativity at each node in  $X_i^N \subset \tau_i$ , Xing et al. use a simple scaling of the DG solution around the non-negative cell average, i.e.  $\mathbf{U}_i^n := \mathbf{U}_h^n|_{\tau_i}$  is modified to  $\tilde{\mathbf{U}}_i^n$  by

$$\tilde{\mathbf{U}}_i^n(\mathbf{x}) = \bar{\mathbf{U}}_i^n + \theta (\mathbf{U}_i^n(\mathbf{x}) - \bar{\mathbf{U}}_i^n), \quad \theta = \min \left\{ 1, \frac{\bar{H}_i^n}{\bar{H}_i^n - \min_{\mathbf{x} \in X_i^N} H_i^n(\mathbf{x})} \right\}. \quad (4.16)$$

In this context, equation (4.16) is called the *positivity-preserving limiter*. This limiter is locally mass and momentum conserving and only requires the positivity of water height averages.

The simulation of wetting and drying using higher order time integration hinges on the subclass of SSP-RK schemes introduced in the beginning of Chapter 3. Thereby, the assertion of Theorem 4.2 may be extended to higher order time integration, see e.g. [214, 220]. In fact, using the positivity preserving limiter, we can ensure that  $H_h^n(\mathbf{x})$  is non-negative at each quadrature point in  $X_i^N$ . Theorem 4.2 then ascertains that the FE assumption is fulfilled for the convex function  $\|\mathbf{U}\| = \max_i \{-\bar{H}_i\}$ . Hence, the resulting DG scheme using SSP-RK time integration preserves non-negativity of the cell means of water height as long as the respective time step constraint is satisfied and the positivity preserving limiter is applied at each stage. Denoting by  $\Delta t_{FE}$  the time step of the forward Euler scheme fulfilling the stability constraints of Theorem 4.2 and defining  $c = \min_{i,j} \frac{\alpha_{ij}}{\beta_{ij}}$  based on the coefficients of the given SSP-RK scheme as defined in (3.51), the assertion is summarized in the following Theorem.

**Theorem 4.4.** *For the DG scheme with positivity preserving numerical flux function and SSP-RK time integration, the new cell averages  $\bar{H}_i^{n+1}$  of the water height at time  $t^{n+1}$  are non-negative under the premises that the time step  $\Delta t$  fulfills the restriction*

$$\Delta t \leq c\Delta t_{FE}$$

*and that the positivity-preserving limiter is applied at each stage of the SSP-RK scheme to obtain non-negative nodal water height in the set  $X_i^N \subset \tau_i$  on all relevant time levels.*

Many of the more recent schemes discussed in Section 4.3.2 follow this approach using SSP-RK schemes when supplementing their respective space discretization by the time stepping routine. In summary, the resulting list of construction principles for methods of this kind consists in:

1. designing a DG scheme for which the FE assumption (3.50) is fulfilled for the above example of a convex function regarding positivity,
2. asserting non-negativity of the cell averages of water height by satisfying the time step constraint of the respective SSP-RK method, given by the parameter  $c$ ,
3. applying the positivity preserving limiter at each Runge-Kutta stage to obtain non-negative water height at the intermediate nodes.

### DG schemes based on the pre-balanced formulation

In [53], Duran and Marche extend a previously constructed FV scheme based on the pre-balanced SW equations to a DG discretization. Hereby, the pre-balanced equations reduce the required degree of exactness of the DG quadrature rule to achieve well-balancedness which is otherwise higher than in the standard approach. Their approach is generally in line with the methods of Xing et al. [214, 213], as they also use hydrostatic reconstruction as described in Section 4.3.1 to determine the input values for the numerical flux, the positivity preserving limiter and strong-stability preserving explicit time integration as described in the previous paragraph. Furthermore, the idea of using the pre-balanced equations has been extended to a DG method which uses finite volume subcells in nearly dry regions by Meister and Ortob in [131].

### Implicit time integration for wetting and drying processes in DG schemes

More recently, DG schemes treating wetting and drying for shallow water flow with implicit time integration have been developed. In fact, when explicit time step restrictions become far too prohibitive, e.g. due to grid stiffness, implicit time stepping is the most obvious alternative. In this case, implicit schemes can yield significant speed-up and permit simulations that would be impossible to carry out with explicit methods.

For real applications this substantial increase in efficiency has been documented e.g. by Kärnä et al. in [96]. In that work, the authors observe that porosity methods have the advantage of smooth transitions for wetting and drying areas which increases their compatibility with

implicit time integration schemes. Their approach to wetting and drying is similar to a negative-depth algorithm but does not need to introduce the concept of porosity since it temporarily moves the bottom topography such that the new surface elevation is always positive. After a DG discretization, the numerical solution is obtained by high order diagonally implicit Runge-Kutta (DIRK) schemes, whereby the nonlinear systems are solved via Newton iteration using a finite-difference approximation to the Jacobian. Due to the removal of switches and discontinuities within the algorithm, the Newton solver is robust and converges rapidly. Difficulties are reported for the third Balzano test case, where the constant presence of the gravitational forces causes the interior pond to dry out. However, this is as a common drawback in porous media methods [141] and not specific to this scheme. Generally, lakes may be artificially emptied when the water surface is aligned with the non-constant bottom topography, creating an artificial flux at the corresponding cell boundaries which moves water out of the lake.

To deal with this problem, in the already mentioned recent work by Le et al. [106], the authors use a thin layer approach within a DG semi-discretization and employ implicit time stepping but avoid to completely cancel out artificial gravity effects in partially dry cells. This approach achieves fast Newton convergence analogous to porosity methods but prevents drying out of interior lakes. Furthermore, the method is able to provide realistic simulations of the Tonle Sap Lake in the Mekong River Basin, which is subject to significant variations of the water level between the dry and the wet season.

In [132], Meister and Ortlob extend the framework of high order SSP explicit time integration devised by Xing et al. [214] to unconditionally positive implicit time integration via the Patankar approach. This method will be elaborated in more detail in Section 4.4.

Furthermore, Marras et al. [125] incorporate a wetting and drying strategy into a unified continuous/discontinuous Galerkin (CG/DG) scheme with dynamically adaptive viscosity as artificial dissipation used for shock capturing. In their work, they extend the already mentioned strategy by Xing, Zhang and Shu for wetting and drying to their CG/DG method and use a three-stage, second order ESDIRK scheme to advance the numerical solution in time. The scheme belongs to the category of fixed grid thin-layer methods. As the first stage of the ESDIRK time integration scheme is explicit and equals the last stage of the previous step, effectively only computations of a two-stage scheme are carried out. The non-linear systems are solved by a Jacobian-free Newton-Krylov scheme, where the GMRES method is used to solve the linear system in each Newton step. This basic strategy is similar to the ones used in the works [96] and [132]. A closer look at the formulation of the governing equations in [125] reveals that a very similar form to the pre-balanced SW equations is used in this work, the only difference given in the use of  $H$  as conserved variable.

## 4.4 Wetting and drying treatment based on the Patankar trick

A positivity preserving and well-balanced DG scheme on unstructured triangular grids for the SW equations was developed by Xing and Zhang in [213], based on hydrostatic reconstruction, positivity preserving numerical fluxes and explicit SSP-RK time integration. Unfortunately,

there are limits to the efficiency of explicit time integration schemes in case of stiff problems. Especially for high polynomial degrees, linear stability requires very small time steps. In addition, since time step restrictions depend on the cell sizes, implicit time stepping is often more efficient for locally refined grids, e.g. due to refinement in wetting and drying regions.

However, also in case of implicit time integration, the positivity preserving approach of Xing et al. [214, 213] needs to enforce non-negative cell means of water height under rather restrictive time step constraints that also depend on the order of the DG discretization. Since these positivity enforced time step restrictions interfere with the efficiency of the implicit time integrator, unconditionally positive implicit schemes are desired.

In order to guarantee non-negativity of the water height for any time step size while still preserving conservation properties, Meister and Ortob [132] modify the strategy of positivity preservation in [214, 213] by the Patankar approach introduced in Section 3.4. Hereby, an implicit time integrator is desired to avoid severe time step size restrictions in case of small grid cells. Thus, time integration is now carried out by the modified Patankar-SDIRK scheme constructed in Section 3.4.3 which is based on an L-stable implicit Runge-Kutta scheme. We recall that the Patankar approach allows for unconditional positivity whereas classical linear schemes, also implicit ones, are either subject to time step restrictions in order to guarantee positivity or are reduced to first order accuracy.

For the purpose of applying the MPSDIRK3 scheme, a suitable production-destruction equation for the cell averages of water height is first extracted from the semi-discrete continuity equation given by the DG scheme. The weights introduced by the Patankar scheme are thereby designed to reduce the outgoing water fluxes which constitute the destruction terms. In the same way, the basic idea of the modified Patankar scheme, i.e. applying corresponding weights to the production terms, yields an analogous modification of the ingoing water fluxes and thereby respects the mass conservation property of the shallow water equations.

#### 4.4.1 Production-destruction splitting of the DG-discretized SW equations

To obtain a suitable production-destruction splitting representing ingoing and outgoing water fluxes, we now take a closer look at the DG discretization for the cell means of the water height  $H$ , given by  $\bar{H}_i = \frac{1}{|\tau_i|} \int_{\tau_i} H_h(\mathbf{x}) d\mathbf{x}$  on the cell  $\tau_i$ .

Neglecting boundary terms, inserting the test function  $\mathbf{W} = (1, 0, 0)^T$  into the DG scheme (4.13), where  $\mathbf{F}^*$  is given by the well-balanced flux  $\mathbf{F}^{WB}$  in (4.14) which modifies a given positivity preserving flux  $\mathbf{F}^{num}$  in order to preserve lake at rest steady states, we have

$$\frac{d}{dt}(|\tau_i| \bar{H}_i) = - \sum_{j \in N(\tau_i)} \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, \mathbf{n}) d\sigma = \sum_{j \in N(\tau_i)} p_{ij} - \sum_{j \in N(\tau_i)} d_{ij}, \quad (4.17)$$

where  $N(\tau_i)$  denotes the set of indices of neighbors to  $\tau_i$  with common edge  $\Gamma_{ij} = \tau_i \cap \tau_j$ ,

$F_1^{num}$  is the first component of  $\mathbf{F}^{num}$  and the terms  $p_{ij}$  and  $d_{ij}$  are given by

$$\begin{aligned} p_{ij} &= \max \left\{ 0, - \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, \mathbf{n}) d\sigma \right\}, \\ d_{ij} &= \max \left\{ 0, \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, \mathbf{n}) d\sigma \right\}. \end{aligned}$$

In this way, we now distinguish between positive and negative flux contributions over element boundaries. The choice for the production and destruction terms hereby guarantees the two properties

$$p_{ij} - d_{ij} = - \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^-, \mathbf{U}_{i,*}^+, \mathbf{n}) d\sigma,$$

implying that the combination of production and destruction terms recovers the entire flux between two adjacent cells, and

$$p_{ij} = d_{ji},$$

signifying conservation of the kind that the amount of water flowing into cell  $\tau_i$  over the edge  $\Gamma_{ij} = \tau_i \cap \tau_j$  is exactly the amount of water leaving the cell  $\tau_j$  through the same edge.

This decomposition of fluxes in positive and negative contributions was also considered by Bollermann et al. [23] for finite volume evolution Galerkin methods. In that work, the authors suggest a different approach to guarantee positivity preservation which is restricted to explicit time integration.

The application of the modified Patankar approach is now based on the resemblance of the derived production-destruction formulation (4.17) to those production-destruction equations frequently encountered in the context of geobiochemical models. Designing a first unconditionally positive time integration method for the cell averages of water height, we could start by considering the modified Patankar scheme (3.55). For the cell means of water height, the fully discrete scheme based on the DG space discretization has the form

$$|\tau_i| \bar{H}_i^{n+1} = |\tau_i| \bar{H}_i^n + \Delta t \left( \sum_{j \in N(\tau_i)} p_{ij}^n \frac{\bar{H}_j^{n+1}}{\bar{H}_j^n} - \sum_{j \in N(\tau_i)} d_{ij}^n \frac{\bar{H}_i^{n+1}}{\bar{H}_i^n} \right), \quad (4.18)$$

where

$$p_{ij}^n = \max \left\{ 0, - \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^{n,-}, \mathbf{U}_{i,*}^{n,+}, \mathbf{n}) d\sigma \right\} = d_{ji}^n.$$

Due to the balanced weights for production and destruction terms, this scheme respects mass conservation as well.

However, since this Patankar modification is based on an explicit Runge-Kutta scheme, stability requirements will still demand a time step restriction depending on the cell sizes. As already mentioned, this can be disadvantageous in case of inhomogeneous grids containing very small cells. Furthermore, since the modified Patankar-Euler scheme is only first order accurate, the construction of higher order time integration schemes based on the Patankar approach will be considered in the following.

#### 4.4.2 MPSDIRK3 time integration for the DG-discretized SW equations

Our aim is to apply the MPSDIRK3 scheme constructed in Section 3.4.3 to the DG-discretized shallow water equations. Regarding the physical quantities, the preservation of non-negativity only concerns the water height. In addition, based on non-negative cell averages  $\bar{H}_i^n$ , the positivity preserving limiter (4.16) by Xing et al. may be applied to enforce non-negative water height at specific nodal points of the DG cells. Therefore, effectuating non-negative cell averages of water height should be the objective of the Patankar approach. Thereby, we achieve non-negative cell means  $\bar{H}_i$  after an intermediate or after the final RK stage of the underlying SDIRK scheme. Regarding the benefit of the positivity preserving limiter, we recall that non-negative water height at the predetermined points on the DG cells is a basic prerequisite specified in Theorem 4.2 to guarantee non-negative cell means  $\bar{H}_i^{n+1}$  after explicit Euler time integration with a sufficiently small time step. In turn, according to Lemma 3.11, we need this property regarding explicit Euler time integration, in conjunction with solvability of the non-linear equations given by an implicit Euler step, in order to guarantee unconditional positivity of the implicit Euler scheme,

Although in general, we will not be able to prove solvability of the implicit Euler scheme applied to the DG semi-discretization of the non-linear shallow water equations if arbitrarily large time steps are taken, numerical experiments demonstrate non-negative cell means of water height for implicit Euler time integration as long as the Newton process solving the resulting system of non-linear equations is convergent. In the following, we will therefore assume that implicit Euler integration preserves non-negativity of the cell averages of water height if the positivity preserving limiter has been applied to the current numerical solution.

The semi-discrete DG scheme in variational formulation (4.13) represents a system of ordinary differential equations which can compactly be written as

$$\mathbf{U}'(t) = \mathbf{g}(t, \mathbf{U}(t)),$$

where the vector  $\mathbf{U}$  now collects the complete set of degrees of freedom of the spatial discretization. For example, in case of a representation via PKD basis functions (1.77) used on triangular cells in Section 1.2.5, the semi-discrete solution  $\mathbf{U}$  is composed of the coefficient vectors  $\hat{\mathbf{u}}_{i,lm}$  on each triangular element  $\tau_i \in \mathcal{T}^h$ .

The MPSDIRK3 algorithm introduced in Section 3.4.3 consists of three stages whereby the first two stages correspond to the unmodified SDIRK3 scheme by Cash specified in Table 3.15. In pseudocode, an unmodified step of the SDIRK3 method is reviewed in Algorithm 4.1.

As already discussed in Section 3.4.3, the MPSDIRK3 algorithm consists in a modification of the third stage of the SDIRK3 scheme since negative cell averages of water height may potentially occur in the sum of lower stages

$$\mathbf{s} = \mathbf{U}^n + \alpha \mathbf{r}^{(1)} + \beta \mathbf{r}^{(2)} = \mathbf{U}^n + \alpha \Delta t \mathbf{g} \left( t^n + \gamma \Delta t, \mathbf{U}^{(1)} \right) + \beta \Delta t \mathbf{g} \left( t^n + (\gamma + \delta) \Delta t, \mathbf{U}^{(2)} \right),$$

computed in Line 6 of Algorithm 4.1.

However, in the present context of time stepping for the DG-discretized shallow water equations, we extend the approach in Section 3.4.3 by constructing a hybrid method. More precisely, we will only use the Patankar modification if time integration using the unmodified

**Algorithm 4.1** Unmodified step of SDIRK3**Input:**  $\mathbf{U}^n, \Delta t$ **Output:**  $\mathbf{U}^{n+1}$ 

- 1: Solve  $\mathbf{U}^{(1)} = \mathbf{U}^n + \gamma \mathbf{g}(t^n + \gamma \Delta t, \mathbf{U}^{(1)})$
- 2:  $\mathbf{r}^{(1)} \leftarrow \Delta t \mathbf{g}(t^n + \gamma \Delta t, \mathbf{U}^{(1)})$
- 3:  $\mathbf{s} \leftarrow \mathbf{U}^n + \delta \mathbf{r}^{(1)}$
- 4: Solve  $\mathbf{U}^{(2)} = \mathbf{s} + \gamma \Delta t \mathbf{g}(t^n + (\delta + \gamma) \Delta t, \mathbf{U}^{(2)})$
- 5:  $\mathbf{r}^{(2)} \leftarrow \Delta t \mathbf{g}(t^n + (\delta + \gamma) \Delta t, \mathbf{U}^{(2)})$
- 6:  $\mathbf{s} \leftarrow \mathbf{U}^n + \alpha \mathbf{r}^{(1)} + \beta \mathbf{r}^{(2)}$
- 7: Solve  $\mathbf{U}^{n+1} = \mathbf{s} + \gamma \Delta t \mathbf{g}(t^n + \Delta t, \mathbf{U}^{n+1})$

SDIRK scheme actually yields at least one negative cell average  $\bar{H}_i$  contained in the above vector  $\mathbf{s}$ .

**Modification of the water height averages**

Our goal is to modify the vector  $\mathbf{s}$  in the third stage, having potentially negative cell averages  $\bar{H}_i^{\mathbf{s}}$ , by a vector  $\mathbf{z}$  having non-negative cell averages of water height  $\bar{H}_i^{\mathbf{z}}$ . For this purpose, we recompute the cell averages of water height using the MPSDIRK3 approach. For simplicity of presentation, boundary terms corresponding e.g. to inflow or outflow conditions or fixed walls are hereby neglected in the formulation below. The inclusion of domain boundary conditions into the formulation is straightforward and can be found in [132].

On interior DG cells, similar to the modified Patankar-Euler approach (4.18), we have

$$\bar{H}_i^{\mathbf{z}} = \bar{H}_i^n + \frac{\Delta t}{|\tau_i|} \sum_{j \in N(\tau_i)} \left[ \alpha \left( p_{ij}^{(1)} \frac{\bar{H}_j^{\mathbf{z}}}{\tilde{H}_j^{(1)}} - d_{ij}^{(1)} \frac{\bar{H}_i^{\mathbf{z}}}{\tilde{H}_i^{(1)}} \right) + \beta \left( p_{ij}^{(2)} \frac{\bar{H}_i^{\mathbf{z}}}{\tilde{H}_i^{(2)}} - d_{ij}^{(2)} \frac{\bar{H}_j^{\mathbf{z}}}{\tilde{H}_j^{(2)}} \right) \right], \quad (4.19)$$

where the fluxes over element boundaries depending on the first two stages  $\mathbf{U}^{(k)}$ ,  $k = 1, 2$ , yield the preliminary production and destruction terms defined by

$$p_{ij}^{(k)} = \max \left\{ 0, - \int_{\Gamma_{ij}} F_1^{num}(\mathbf{U}_{i,*}^{(k),-}, \mathbf{U}_{i,*}^{(k),+}, \mathbf{n}) d\sigma \right\} = d_{ji}^{(k)}, \quad k = 1, 2, \quad (4.20)$$

and the denominators  $\tilde{H}_i^{(k)}$  of the Patankar weights are given by

$$\tilde{H}_i^{(k)} = \begin{cases} \bar{H}_i^{\mathbf{s}}, & \text{if } \bar{H}_i^{\mathbf{s}} > H_\epsilon, \\ \bar{H}_i^{(k)}, & \text{otherwise,} \end{cases} \quad (4.21)$$

where a suitable positive tolerance  $H_\epsilon$  of the water height needs to be chosen.

Now, the scheme (4.19) is based on the assumption that the quantities  $\frac{p_{ij}^{(1)}}{\tilde{H}_j^{(1)}} = \frac{d_{ji}^{(1)}}{\tilde{H}_j^{(1)}}$  and  $\frac{p_{ij}^{(2)}}{\tilde{H}_i^{(2)}} = \frac{d_{ji}^{(2)}}{\tilde{H}_i^{(2)}}$  are well-defined also for vanishing water height. Hence, in analogy to the computation

of damped velocities in case of a very small water height in [105, 214, 27] and equation (4.12), the preliminary production and destruction terms are furthermore modified to

$$\frac{p_{ij}^{(k)}}{\tilde{H}_l^{(k)}} = \frac{d_{ji}^{(k)}}{\tilde{H}_l^{(k)}} = \begin{cases} 0, & \text{if } \tilde{H}_l^{(k)} < H_\epsilon, \\ 2\tilde{H}_l^{(k)} \cdot p_{ij}^{(k)} / \left( \left( \tilde{H}_l^{(k)} \right)^2 + \max \left\{ \left( \tilde{H}_l^{(k)} \right)^2, H_\epsilon \right\} \right), & \text{otherwise,} \end{cases} \quad (4.22)$$

where  $l = j$  for  $k = 1$  and  $l = i$  for  $k = 2$ .

**Remark 4.5.** By keeping the cell averages  $\bar{H}_i^s$  as Patankar weight denominators for moderate values corresponding to a thin layer of water height  $\bar{H}_i^s > H_\epsilon$ , we intend to reduce the influence of the Patankar modification in wet areas away from wet/dry fronts as theoretically explained in Section 3.4.3 for general ODEs of the form (3.52).

### Modification of the remaining degrees of freedom

So far, we have only considered how to modify the cell means of  $H$  to keep these quantities non-negative independent of the time step size. The vector  $\mathbf{z}$  to be constructed has the form

$$\mathbf{z} = \left[ \begin{array}{c} \hat{H}_{i,lm}^{\mathbf{z}} \\ \widehat{(H\mathbf{v})}_{i,lm}^{\mathbf{z}} \end{array} \right]_{i,lm},$$

with  $\hat{H}_{i,oo}^{\mathbf{z}} = \bar{H}_i^{\mathbf{z}}$  already determined by the Patankar approach.

The cell averages of the discharge don't have to be modified, thus we set  $(\widehat{H\mathbf{v}})_{i,oo}^{\mathbf{z}} = (\widehat{H\mathbf{v}})_{i,oo}^s$ . The next step is to adjust the higher order coefficients contained in the vector for all conservative variables including the discharges. In [132], we suggest to cancel the higher order degrees of freedom for vanishing water height, i.e. we set

$$\hat{H}_{i,lm}^{\mathbf{z}} = (\widehat{Hv_1})_{i,lm}^{\mathbf{z}} = (\widehat{Hv_2})_{i,lm}^{\mathbf{z}} = 0 \text{ for } l + m > 0 \quad \text{if } \bar{H}_i^s < 0. \quad (4.23)$$

Furthermore, in order to obtain non-negative cell means  $\bar{H}_i$  after the next RK stage to be executed, we have to employ the positivity preserving limiter (4.16) by Xing et al. again to adjust the values of  $H$  at the set of nodes  $X_j^N$ . We recall that this procedure is necessary to enforce the forward Euler assumption (3.50) which enables us to exploit the unconditional SSP property of the backward Euler scheme. Applying the positivity preserving limiter to a vector  $\mathbf{z}$  will be denoted by  $\text{PPlim}(\mathbf{z})$ .

With the newly computed vector  $\mathbf{z}$  substituted for  $\mathbf{s}$  in Line 7 of Algorithm 4.1, we may therefore again assume that the solution vector  $\mathbf{U}^{n+1}$  obtained by the third stage which corresponds to an implicit Euler type step with initial data  $\mathbf{z}$  and time step  $\gamma\Delta t$  contains only non-negative water height averages.

### Accuracy away from the wet/dry front

Although the global accuracy of the MPSDIRK3 scheme reduces to first order in case of vanishing water height, the specific choice of Patankar weight denominators in (4.21) principally

allows for a higher order of accuracy away from the wet/dry front. In particular, transferred to the context of the DG-discretized SW equations, conditions (C1)–(C3) on page 195 in Section 3.4.3 are fulfilled for a specific cell  $\tau_i$  if the water height average  $\bar{H}_i^n$  and the averages  $\bar{H}_j^n$  on triangles  $\tau_j$  either adjacent to  $\tau_i$  or adjacent to neighbors of  $\tau_i$  are larger than the thin layer tolerance  $H_\epsilon$  in (4.21). For a typical triangular cell  $\tau_i$  within an unstructured grid, this neighborhood described above is also depicted in Figure 4.5.

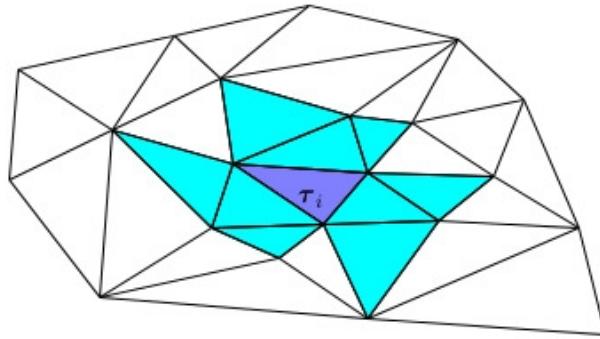


Figure 4.5: Neighborhood to a triangle  $\tau_i$ , in which the order of the scheme is affected by a possible wet/dry transition in  $\tau_i$ .

That is, if a large enough neighborhood of  $\tau_i$  lies in a wet region away from the wet/dry front, we can expect third order accuracy of the time integration scheme in this region. On the other hand, in wet/dry transition areas, we can expect at most first order accuracy of the time integration scheme. However, this is in line with the approximation order in space, since dissipation mechanisms such as modal filtering described in Section 4.4.3 often locally reduce the accuracy of the spatial discretization at the shoreline to first order as well.

### Summary of the MPSDIRK3 routine for the DG-discretized SW equations

Complementing the MPSDIRK3 scheme by applications of the positivity preserving limiter (4.16), a complete MPSDIRK3 time step for the DG-discretized SW equations is listed in Algorithm 4.2.

#### 4.4.3 Shock capturing by modal filtering

In order to introduce a small but sufficient amount of numerical dissipation to the DG scheme in case of non-smooth solutions, specific damping mechanisms are frequently applied. One approach is to introduce numerical dissipation by modal filtering as described in [134, 135]. We will give a brief review of this filtering procedure in the following.

Modal filtering relies on a modal representation of the DG solution. For this purpose, in case of triangular grids, the approximate solution on each triangle may be represented using the PKD basis functions (1.77) introduced in Section 1.2.5. The approximation  $\mathbf{U}_i = \mathbf{U}_h|_{\tau_i}$  on a specific triangle  $\tau_i$  is then given by the expansion

$$\mathbf{U}_h(\psi_i^{-1}(r, s), t) = \sum_{l+m \leq N} \hat{\mathbf{u}}_{i,lm}(t) \Phi_{lm}(r, s),$$

---

**Algorithm 4.2** MPSDIRK3 step for DG-discretized SW equations

---

**Input:**  $\mathbf{U}^n$  containing only non-negative water height averages,  $\Delta t$   
**Output:**  $\mathbf{U}^{n+1}$  containing only non-negative water height averages

- 1:  $\mathbf{U}^n \leftarrow \text{PPlim}(\mathbf{U}^n)$
- 2: Solve  $\mathbf{U}^{(1)} = \mathbf{U}^n + \gamma \mathbf{g}(t^n + \gamma \Delta t, \mathbf{U}^{(1)})$
- 3:  $\mathbf{r}^{(1)} \leftarrow \Delta t \mathbf{g}(t^n + \gamma \Delta t, \mathbf{U}^{(1)})$
- 4:  $\mathbf{s} \leftarrow \mathbf{U}^n + \delta \mathbf{r}^{(1)}$
- 5:  $\mathbf{s} \leftarrow \text{PPlim}(\mathbf{s})$
- 6: Solve  $\mathbf{U}^{(2)} = \mathbf{s} + \gamma \Delta t \mathbf{g}(t^n + (\delta + \gamma) \Delta t, \mathbf{U}^{(2)})$
- 7:  $\mathbf{r}^{(2)} \leftarrow \Delta t \mathbf{g}(t^n + (\delta + \gamma) \Delta t, \mathbf{U}^{(2)})$
- 8:  $\mathbf{s} \leftarrow \mathbf{U}^n + \alpha \mathbf{r}^{(1)} + \beta \mathbf{r}^{(2)}$
- 9: **if** there is  $i$  with  $\bar{H}_i^{\mathbf{s}} < 0$  **then**
- 10:    $\mathbf{U}^{(1)} \leftarrow \text{PPlim}(\mathbf{U}^{(1)})$
- 11:    $\mathbf{U}^{(2)} \leftarrow \text{PPlim}(\mathbf{U}^{(2)})$
- 12:   Compute  $\frac{p_{ij}^{(k)}}{\bar{H}_i^{(k)}}, \frac{d_{ji}^{(k)}}{\bar{H}_i^{(k)}}$  by (4.20) and (4.22)
- 13:    $\mathbf{z} \leftarrow \mathbf{s}$
- 14:   Compute  $\bar{H}_i^{\mathbf{z}}$  by (4.19)
- 15:   Replace water height averages of  $\mathbf{z}$  by  $\bar{H}_i^{\mathbf{z}}$  and modify higher order coefficients by (4.23)
- 16:    $\mathbf{s} \leftarrow \mathbf{z}$
- 17:    $\mathbf{s} \leftarrow \text{PPlim}(\mathbf{s})$
- 18: **end if**
- 19: Solve  $\mathbf{U}^{n+1} = \mathbf{s} + \gamma \Delta t \mathbf{g}(t^n + \Delta t, \mathbf{U}^{n+1})$

---

where  $\psi_i$  maps the specific triangle to the reference element, see (1.78).

In [134, 135], exponential filters including a shock indicator to adapt artificial dissipation to shock positions have been developed. Modal filters of this kind are function  $\sigma : [0, 1] \rightarrow [0, 1]$  of the form

$$\sigma(\mu) = \exp(-\alpha_i s_i \mu^{2p}), \quad (4.24)$$

with the shock indicator  $s_i$ , the filter order  $2p$ , and the filter strength  $\alpha_i = C_p \frac{N \Delta t}{h_i}$ , where  $N$  is the polynomial degree of the DG representation,  $C_p$  is a constant and  $\Delta t$  and  $h_i$  denote the time step size and the shortest height of  $\tau_i$ , respectively. The parameter  $\mu$  refers to the specific modal coefficient in the modal representation of the DG solution, since the aim of modal filtering is to apply an increasing amount of artificial dissipation to higher order coefficients in comparison to the numerical diffusion applied to lower order ones. Exponential modal filtering by a filter function of the form (4.24) thus modifies the DG approximation to

$$\mathbf{U}_h^{\text{mod}}(\psi_i^{-1}(r, s), t) = \sum_{l+m \leq N} \hat{\mathbf{u}}_{i,lm}^{\text{mod}}(t) \Phi_{lm}(r, s),$$

where

$$\hat{\mathbf{u}}_{i,lm}^{\text{mod}} = \exp(-\alpha_i s_i \eta^{2p}) \hat{\mathbf{u}}_{i,lm}, \quad \eta = \frac{l+m}{N+1}, \quad (4.25)$$

is the modified vector of PKD coefficients.

In [133], the filter action on the water height representation  $H_h$  was modified in order to preserve the property of well-balancedness. However, further investigations showed, that it is sufficient to rely on a well-balanced definition of the shock indicator  $s_i$  in (4.25). This indicator is based on the decay rate of PKD coefficients, similar to the definition in [159], where now the coefficients of the water surface level are taken into account to guarantee well-balancedness. The precise form of the shock indicator is given by

$$s_i = (\bar{H}_i^*)^{-1} \min \left\{ 1, 1000(5N^4 + 1) \sum_{l+m=N} (\hat{\eta}_{i,lm})^2 \cdot \left( \sum_{l+m < N} (\hat{\eta}_{i,lm})^2 + \tilde{\epsilon} \right)^{-1} \right\}, \quad (4.26)$$

where  $\tilde{\epsilon} = 10^{-10}$  is a small regularization parameter, and  $\hat{\eta}_{i,lm} = \hat{H}_{i,lm} + \hat{b}_{i,lm}$  denotes the coefficients of surface elevation. In order to introduce more stability in nearly dry regions, a division by the cell mean of water height is introduced in the definition (4.26), different from the resolution indicator defined in [133]. Hereby,  $\bar{H}_i^* = \max\{\bar{H}_i, H_\epsilon\}$  denote the cell means of water height cut above the thin layer tolerance  $H_\epsilon$ .

#### 4.4.4 Numerical experiments

In the following, we report the behavior of the MPSDIRK3 scheme for the DG-discretized shallow water equations with respect to three selected test cases computed in [132].

Before describing the specific results, some of the aspects of the numerical algorithm need clarification. First, the thin layer tolerance  $H_\epsilon$  to be defined by the user is set to  $H_\epsilon = 10^{-6}$

throughout this section. Furthermore, as already discussed, a direct computation of the velocities as  $\mathbf{v}_i = (H\mathbf{v})_i / H_i$  will eventually lead to numerical instabilities for small water height even for bounded discharges. Similar to regularization (4.12) first introduced by Kurganov and Petrova, we therefore compute damped velocities within the DG scheme whenever point-wise values are required. More precisely, we set

$$\mathbf{v}_i = \mathbf{0} \text{ if } H_i < H_\epsilon \text{ and } \mathbf{v}_i = \frac{2H_i \cdot (H\mathbf{v})_i}{H_i^2 + \max\{H_i^2, H_\epsilon\}} \text{ otherwise.}$$

The gravitational constant was set to  $g = 9.812$  in all the following examples and as the positivity preserving numerical flux function  $\mathbf{F}^{num}$  in (4.14), we use the HLL flux described in [76].

For comparison to the implicit MPSDIRK3 time integration scheme we also applied the explicit third order Shu-Osher TVD Runge-Kutta time discretization (TVD-RK3) developed in [177]. The nonlinear systems of equations arising due to the implicit time integration were solved using a Jacobian-free Newton-GMRES scheme. In order to prevent the error in time to dominate the error in space, the CFL number for implicit time integration was set quite low, in fact we set

$$\Delta t = h_i \cdot \max\{|\mathbf{v}_i| + \sqrt{gH_i}\}$$

unless otherwise specified, where  $h_i$  denotes the shortest height of the triangle  $\tau_i$ . The time step choice for the explicit scheme was based on the positivity requirement leading to smaller time steps than the linear stability requirement alone.

### Oscillating Lake

For an illustration of the properties of both implicit and explicit scheme in the context of wetting and drying, we first study a test case proposed in [60] which simulates an oscillating lake in a paraboloidal vessel depicted in Figure 4.6. Hereby, we consider the computational domain  $\Omega = [-2, 2] \times [-2, 2]$ , and the bottom elevation given by

$$b(\mathbf{x}) = 0.1(x_1^2 + x_2^2).$$

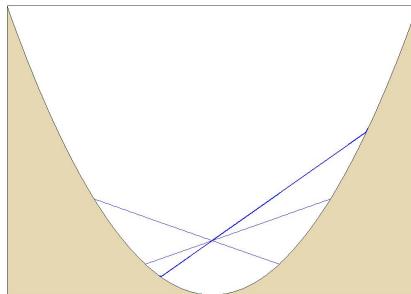


Figure 4.6: 2D cut of paraboloidal vessel at  $y = 0$ .

A periodic analytical solution of the shallow water equations (4.1) is given by

$$\begin{aligned} H(\mathbf{x}, t) &= \max\{0, 0.05(2x_1 \cos(\omega t) + 2x_2 \sin(\omega t)) + 0.075 - b(\mathbf{x})\}, \\ v_1(\mathbf{x}, t) &= -0.5 \omega \sin(\omega t), \\ v_2(\mathbf{x}, t) &= 0.5 \omega \cos(\omega t), \end{aligned}$$

where  $\omega = \sqrt{0.2g}$ .

This test case examines the ability of the numerical methods to deal with wetting and drying. Therefore, it also tests the suitability of the positivity preserving limiter in combination with modal filtering. As initial conditions for the DG scheme, the values of the analytical solution at  $t = 0$  were taken. As the fluid never reaches the boundary of the computational domain, the choice of boundary conditions is less crucial. In our computations, periodic boundary conditions were implemented. The algorithm was carried out for a polynomial degree of  $N = 2$  on a computational grid consisting of  $K = 23138$  elements. Filter parameters were set to  $p = 1$  and  $C_p = 10$ .

Figure 4.7 presents a 3D view of the water surface elevation  $\eta = H + b$  and the discharges  $Hv_1, Hv_2$  of the DG solution at output time of  $T = T_{per}/6$ , where  $T_{per} = 2\pi/\omega$  is the oscillation period. Hereby, coloring refers to the non-flat bottom topography  $b$ . Figure 4.8 depicts water height  $H$  and discharges  $Hv_1, Hv_2$  of the DG solution at a late output time of  $T = 5 \cdot T_{per}$ . Hereby, the implicit MPSDIRK3 scheme described in Algorithm 4.2 was used for time integration. Due to the large number of cells, only the cell means were used for visualization. In Figure 4.9 the time evolution of the wet/dry transition zone is shown for output times of  $T = T_{per}/6, T_{per}/3, T_{per}/2, T_{per}, 2 \cdot T_{per}, 5 \cdot T_{per}$ . The results agree very well with those presented in [214]. Hence, one can conclude that the wetting and drying treatment suggested in that work may also be combined with shock capturing by modal filtering and implicit time integration.

In Table 4.1, we compare the CPU times of our MPSDIRK3 scheme to those obtained by the TVD-RK3 scheme of Shu and Osher for the oscillating lake test on increasingly stiff computational grids. The stiffness of the grids is increased via local refinement as depicted in Figure 4.10, where we use the stiffness measure  $S = \frac{\max_{\tau_i \in T^h} |\tau_i|}{\min_{\tau_i \in T^h} |\tau_i|}$ . According to the results, the implicit scheme beats the explicit one by a factor up to 3.5. Table 4.1 furthermore lists the mass conservation errors committed by the implicit scheme. Full conservation can obviously only be achieved if the accuracy within the iterative solver is set to zero, which is neglected due to practical reasons as usual. However, the results in Table 4.1 show that the corresponding conservation error can be neglected.

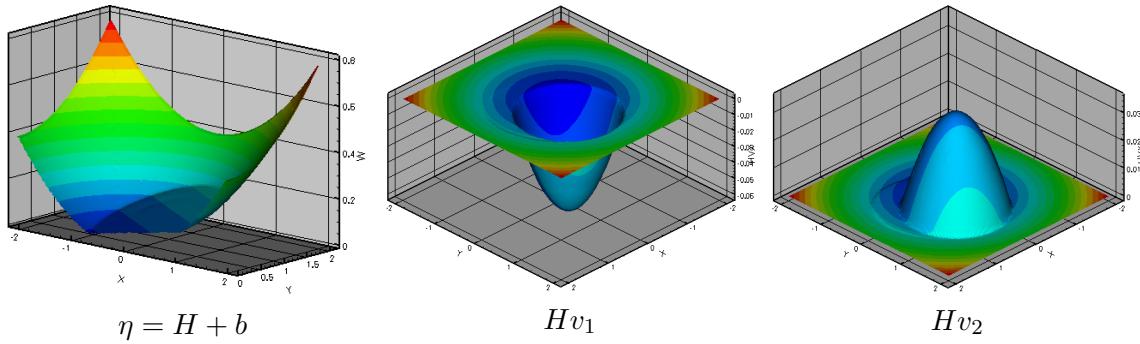


Figure 4.7: Oscillating lake. DG-MPSDIRK3 solution with modal filtering (parameters  $p = 1$  and  $C_p = 10$ ) for a polynomial degree of  $N = 2$  and  $K = 23138$  elements. 3D view of water surface elevation  $\eta = H + b$  and discharges  $Hv_1, Hv_2$  at output time  $T = T_{per}/6$ , only cell means are used for visualization.

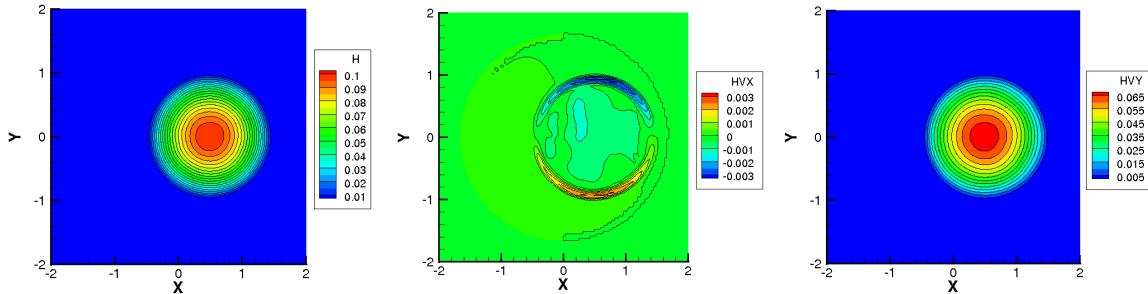


Figure 4.8: Oscillating lake. DG-MPSDIRK3 solution with modal filtering (parameters  $p = 1$  and  $C_p = 10$ ) for a polynomial degree of  $N = 2$  and  $K = 23138$  elements. Water height  $H$  and discharges  $Hv_1, Hv_2$  at output time  $T = 5 \cdot T_{per}$ , only cell means are used for visualization.

Stiffness $S$	Avg. $\Delta t_{EX}$	Avg. $\Delta t_{IM}$	$\frac{CPU_{EX}}{CPU_{IM}}$	$e_{cons}$
6.5	2.99e-4	1.07e-2	0.65	2.31e-14
25.9	1.51e-4	5.42e-3	0.82	1.11e-14
103.4	7.55e-5	2.71e-3	1.29	8.88e-15
413.7	3.77e-5	1.36e-3	1.52	6.93e-14
1654.6	1.89e-5	6.79e-4	1.34	2.25e-13
105894.6	2.40e-6	8.57e-5	3.51	5.42e-13

Table 4.1: CPU time comparison and conservation error of implicit scheme.

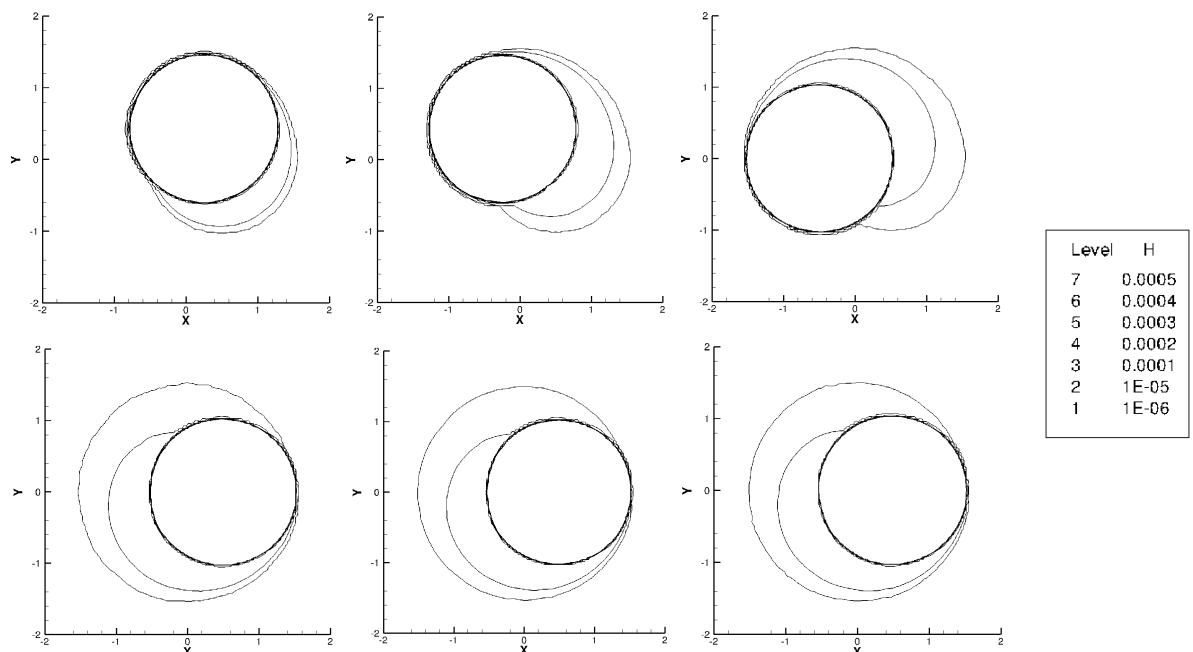


Figure 4.9: Wet/dry transition: Water height levels at  $T = T_{per}/6$ ,  $T_{per}/3$ ,  $T_{per}/2$ ,  $T_{per}$ ,  $2 \cdot T_{per}$  and  $5 \cdot T_{per}$ .

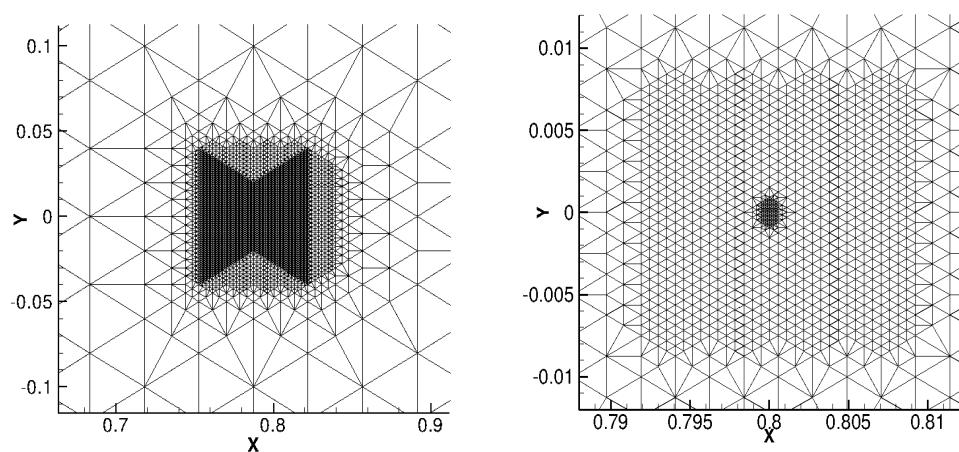


Figure 4.10: Stiff computational grids with  $S = 1654.6$  (left) and  $S = 105894.6$  (right).

### Small Perturbation of a Steady State

This classical test case was given by LeVeque in [113]. It illustrates the combination of the well-balanced DG scheme with modal filtering and implicit MPSDIRK3 time integration. The computational domain is the rectangle  $\Omega = [0, 2] \times [0, 1]$ . The non-constant bottom topography is given by the function

$$b(\mathbf{x}) = 0.8e^{-5(x_1 - 0.9)^2 - 50(x_2 - 0.5)^2}$$

and the initial fluid depth is

$$H(\mathbf{x}, 0) = \begin{cases} 1 - b(\mathbf{x}) + 0.01, & \text{if } 0.05 \leq x_1 \leq 0.15, \\ 1 - b(\mathbf{x}), & \text{otherwise.} \end{cases}$$

Thus, the surface  $H + b$  is almost flat except for the small perturbation by 0.01 for  $0.05 \leq x_1 \leq 0.15$ , also illustrated in Figure 4.11. Furthermore, the velocity is initially set to  $\mathbf{v} = \mathbf{0}$ . The boundary conditions were specified in the following way. As in [27], periodic boundary conditions were implemented at the lower and upper boundaries, although in [113], zero-extrapolation (outflow) conditions were prescribed. At the left and right boundary, we employed outflow conditions as in [113]. However, outflow conditions in combination with numerical dissipation introduced only by modal filters led to an instability of the scheme with large velocities directly at the boundary. However, no problems occurred with periodic conditions on a larger domain. Therefore, we used the TVD limiter by Cockburn and Shu [43] with parameter  $M = 0$  for those cells directly at the computational boundaries.

Figure 4.12 depicts the DG solution with modal filtering ( $p = 1$ ,  $C_p = 10$ ) showing the approximate surface  $w = b + H$  at different output times  $T$  for a polynomial degree of  $N = 2$  on a computational grid consisting of  $K = 46360$  elements. As for the previous oscillating lake test case, only the cell means were used for visualization. Here, the main purpose is to show that our filtering techniques produce correct results and a very detailed resolution of the small perturbation also in the case of implicit time integration. As in the case of explicit time integration shown in [133], the basic features of the results are in very good agreement with those presented in [212]. We obtain a better resolution of the wave structures due to less dissipative shock capturing and in comparison with the second-order computations on a much finer grid in [27], the advantage of a higher order scheme is visible.

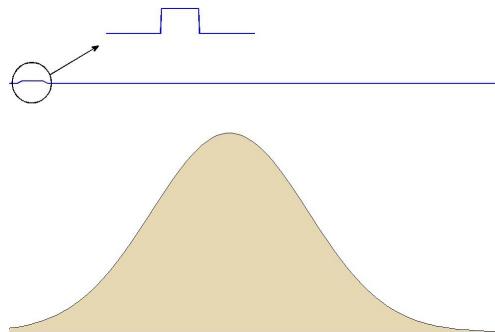


Figure 4.11: 2D cut of the bottom topography and the initial water surface for the small perturbation test case.

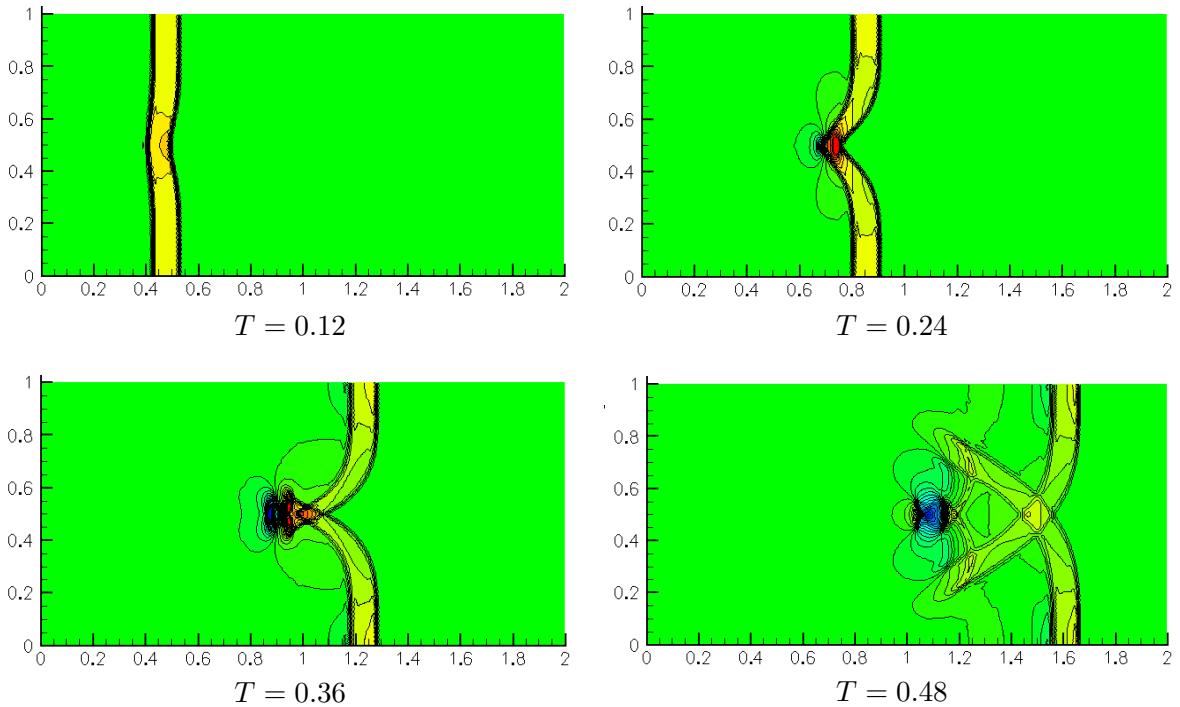


Figure 4.12: Small perturbation test case. Water surface  $\eta = H + b$  output times  $T = 0.12$ ,  $T = 0.24$ ,  $T = 0.36$  and  $T = 0.48$ , with 30 contour levels from 0.99 to 1.01.

### Wetting and Drying on a Sloping Shore

This test case proposed in [188, 168] describes the run-up and successive reflection of a wave on a slope given by the bottom topography

$$b(\mathbf{x}) = b(x_1) = \begin{cases} 0, & \text{if } x_1 < 2x_a, \\ \frac{x_1 - 2x_a}{19.85}, & \text{otherwise,} \end{cases}$$

where  $x_a = \sqrt{\frac{4D}{3\delta}} \operatorname{arcosh} \left( \sqrt{\frac{1}{0.05}} \right)$ ,  $D = 1$ ,  $\delta = 0.019$ . The initial water height and initial velocity vector for this test case are given by

$$\begin{aligned} H_0(\mathbf{x}) = H(\mathbf{x}, 0) &= \max \left\{ D + \delta \operatorname{sech}^2(\gamma(x_1 - x_a)) - b(\mathbf{x}), 0 \right\}, \\ \mathbf{v}_0(\mathbf{x}) = \mathbf{v}(\mathbf{x}, 0) &= \left( \sqrt{\frac{g}{D}} H_0(\mathbf{x}), 0 \right)^T, \end{aligned}$$

where  $\gamma = \sqrt{\frac{3\delta}{4D}}$ . The computational domain for this example is the rectangle  $\Omega = [0, 80] \times [0, 2]$ . The modal filter parameters for this test case were set to  $p = 1$  and  $C_p = 10$ .

Figures 4.14 and 4.15 show the 2D cuts at  $x_2 = 0.05$  of the DG solutions for  $N = 2$  at subsequent output times  $T = 9, 17, 23, 28, 80$ . On the left side, the approximation is obtained by the DG scheme with explicit time integration using a very fine grid for which the computational domain was reduced to  $[0, 80] \times [0, 0.2]$ . This grid then consists of 33602 triangles with minimal and maximal areas of  $|\tau_{min}| \approx 3.53 \cdot 10^{-4}$  and  $|\tau_{max}| \approx 6.06 \cdot 10^{-4}$ , respectively,

and corresponds to a subdivision of the interval  $[0, 80]$  into 2400 elements of equal size at the lower and upper grid boundary. On this fine grid, the approximate solutions using the implicit MPSDIRK3 scheme are visually indistinguishable from those obtained by explicit time integration. Hence, the fine grid solution serves as reference solution for the numerical results on the much coarser grid composed of 348 elements depicted in Figure 4.13 for the region  $\{\mathbf{x} \in \Omega \mid x_1 \in [40, 80]\}$ .

The coarse grid computations, shown on the right side of Figures 4.14 and 4.15, are carried out for a better comparison of implicit and explicit time integration. For  $T = 17$ , the DG solutions show a small overshoot to the right of the dashed bottom line in both cases of explicit and implicit time integration. This reflects the fact that the positivity preserving limiter (4.16) guarantees non-negative water height only at the specified quadrature points but not in the complete triangular subdomain. The numerical results for the coarse grid as well as the visually indistinguishable solutions for the fine grid show that both explicit and implicit scheme have the same ability to correctly reproduce the separate wetting and drying phases as well as the balanced stationary state after long time integration. Although the implicit MPSDIRK3 scheme uses much larger time steps of  $\Delta t = 2h_i \cdot \max\{|\mathbf{v}_i| + \sqrt{gH_i}\}$  on the coarse grid, the results are nonetheless as accurate as those obtained by the explicit TVD-RK3 scheme. Hence, in this prototype example of wetting and drying processes, there is a potential to increase the time step above the stability and positivity restrictions posed by explicit time integration. Furthermore, the implicit scheme yields a better representation of the stationary state at  $T = 80$  as in this case, the coarse grid solution is closer to the straight line of the reference solution.

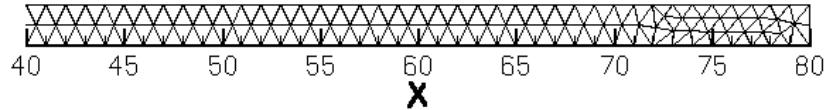


Figure 4.13: Coarse grid used for comparison of implicit vs. explicit. time integration.

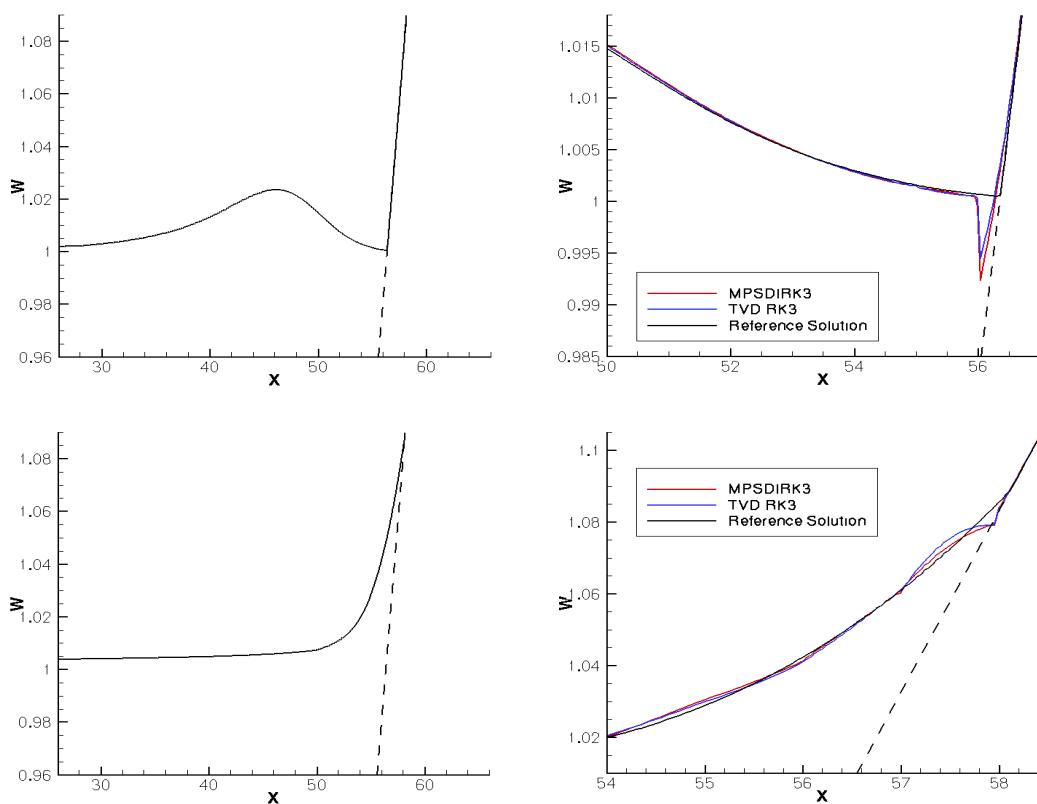


Figure 4.14: Sloping shore. DG solutions for output times from top to bottom:  $T = 9, 17$ ; Water surface  $w = h + H$ . Left side: reference solution, right side: close-up for coarse grid comparison of implicit and explicit time integration.

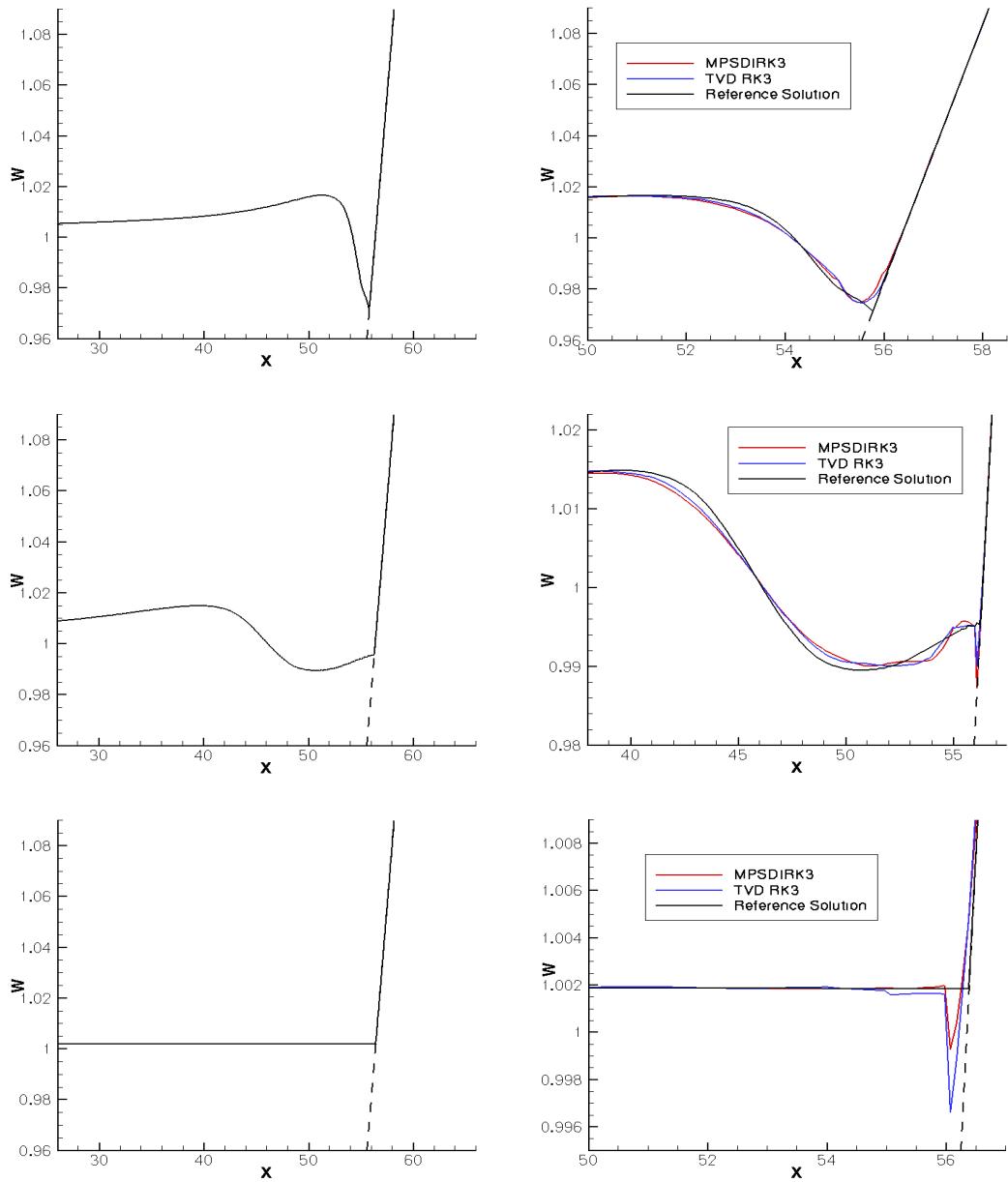


Figure 4.15: Sloping shore. DG solutions for output times from top to bottom:  $T = 23, 28, 80$ ; Water surface  $w = h + H$ . Left side: reference solution, right side: close-up for coarse grid comparison of implicit and explicit time integration.

# Summary and Future Prospects

In this thesis, several aspects regarding the numerical simulation of fluid flow problems were discussed.

Regarding high order methods for space discretization, local and global SBP properties were detected in specific classes of DG schemes and flux reconstruction methods. Hereby, element-wise generalized SBP operators are obtained from nodal DG and flux reconstruction schemes on strictly interior nodal sets. The SBP properties of DG schemes were described in one space dimension, on tensor-product grids and on unstructured triangular grids. On the other hand, considering the DG scheme on a global level and including numerical flux functions results in a global upwind SBP operator. The identified global upwind SBP form of the DG-discretized advection equation should be extendable in principle to nonlinear conservation laws and higher space dimensions. In addition, the familiar subcell finite volume property of DG schemes on Legendre-Gauss-Lobatto nodes was transferred to the DG variant on strictly interior Legendre-Gauss nodes. The DG scheme may thus be regarded from a microscopic view on each cell by its subcell finite volume form and on a macroscopic scale by its global upwind SBP form. While the subcell finite volume form may be used for instance to develop a shock capturing mechanism with subcell resolution, the global upwind SBP form measures the amount of diffusion introduced by the numerical flux functions employed at cell interfaces. This shows that each of these formulations is useful in their own right. Regarding their future prospects, these multifaceted representations of the DG scheme should therefore continue to be of use for the further analysis and design of suitable numerical schemes for fluid flow problems.

The SBP property itself enables to discretize skew-symmetric forms of conservation laws in a manner which both satisfies the primary conservation principles and specific secondary balances. In this work, we therefore used the generalized SBP properties of the DG scheme on Legendre-Gauss-Lobatto nodes to construct kinetic energy preserving Legendre-Gauss DG schemes for the Euler- and Navier-Stokes equations and energy conservative Legendre-Gauss DG schemes for shallow water flow over non-flat bottom topography. When combined with suitable correction terms for cell interfaces which compensate for the lack of cell boundary nodes, these schemes can successfully compete with their counterparts on Legendre-Gauss-Lobatto nodes in terms of accuracy and efficiency. The application of the standard and KEP-DG schemes on both Legendre-Gauss and Legendre-Gauss-Lobatto nodes to the moving piston problem provides an example of a coupled system where the additional KEP property of the fluid solver not only results in a higher accuracy of the fluid solution but also in a better representation of the structure displacement. In this context, the KEP property of the LGL

DG scheme also compensates for the lower degree of exactness of the LGL nodes.

The discussion of energy conservative DG schemes for shallow water flow in this work also contains a comparison to the MaMEC scheme on non-uniform staggered grids in terms of well-balancedness for moving water equilibria. It was shown that while the MaMEC scheme guarantees the preservation of these more general stationary states, the energy conservative DG schemes do not possess this property even though they are well-balanced regarding the lake at rest steady state. Considering the application of SBP schemes to hyperbolic systems of conservation laws which are given in skew-symmetric form on staggered grids could therefore be an interesting route for future investigations.

Dealing with the multitude of available schemes which realize the discretization of diffusion terms in DG framework, specific aspects regarding their interrelation and stability properties were studied. In particular, interrelations were found between  $(\sigma, \mu)$ -schemes which are based on the inclusion of penalty terms and contemporary discretizations using an auxiliary variable for the solution derivative. Furthermore, recent results on energy stability of the BR2 flux employed within nodal DG or ESFR schemes in one space dimension were reviewed and extended. This results in a simplified form of the condition on the BR2 penalty parameter to guarantee energy stability and yields equivalency assertions among different implementations of BR2 schemes with respect to the computation of the BR2 lifting operator on the one hand and of specific BR2 schemes to the BR1 scheme on the other hand.

Subsequently, we studied the influence of various DG diffusion discretizations on dissipation and dispersion properties for advection-diffusion equations. By a comparison of the wave propagation properties of the DG scheme using BR2 diffusion fluxes, a similarity to the results of the BR1 flux for low values of the penalty parameter  $\eta_e$  could be observed. In addition, there is no optimal choice of  $\eta_e$  which provides the most accurate result for all wave numbers and polynomial degrees. In fact, an alternating behavior could be observed, whereby the results for the lowest penalty parameter to guarantee energy stability and a higher value of  $\eta_e = 3$  alternate to provide the smallest error. Considering the alternate LDG fluxes, the performance of  $\text{LDG}_a$  is in general more favorable compared to  $\text{LDG}_b$ , both for the well-resolved problem and the low-resolution test case, independent of the polynomial degree and the nodal DG set. For well-resolved wave numbers, the observation of higher accuracy of the BR1 scheme compared to LDG generally only holds in case of Legendre-Gauss collocation and even polynomial degree of the DG approximate solution. Furthermore, a similar odd-even phenomenon was discovered for DG schemes on Legendre-Gauss nodes applied to the well-resolved problem, since the BR2 schemes with moderate and large penalty parameters beat the BR1 scheme in all odd degree cases, with errors decreasing for increasing  $\eta_e$  while the BR2 scheme for the smallest energy stable penalty parameter performed best in all even degree cases.

Regarding time integration for viscous flow computations, the stability properties of a range of IMEX-DG schemes using advection-diffusion IMEX splitting were investigated. Thereby, a fully discrete  $L^2$ -stability analysis was carried out for the 1D linear advection-diffusion equation discretized in space by the DG scheme using  $(\sigma, \mu)$  diffusion fluxes and in time by IMEX Runge-Kutta schemes. The focus of this investigation was set on the influence of the specific diffusion treatment. With the objective of singling out combinations which allow a time step restriction solely based on the advection and diffusion coefficients and independent

of grid refinement, a sufficient condition regarding the parameters  $\sigma$  and  $\mu$  was found by a theoretical analysis. This condition is fulfilled in particular by the BR2 scheme, by a symmetric form of the LDG scheme on Gauss-Lobatto nodes, and by the recent  $(\frac{1}{4}, \frac{9}{4})$ -recovery scheme. However, the BR1 scheme and the Baumann-Oden method are not unconditionally stable in this sense. This behavior was also demonstrated in corresponding numerical experiments. Consequently, apart from the alternate LDG fluxes which have already been considered in this context, specific  $(\sigma, \mu)$ -schemes also possess the desired stability properties with respect to IMEX time discretization.

This insight is relevant in particular for the a-priori selection of the different building blocks in the design of a numerical scheme, keeping in mind that both variants of the BR scheme are widely used DG diffusion treatments in computational fluid dynamics. If IMEX time integration is to be applied in advection-diffusion split form, the results indicate that a small amount of artificial diffusion concerning viscous flux discretization is necessary and schemes with neutral behavior such BR1 should be excluded in this case. There is also a link to the respective upwind SBP properties of the specific DG diffusion treatments involved in the analysis. While the BR1 scheme is a second-derivative generalized SBP operator not requiring the extension by an upwind characterization, the alternate LDG fluxes represent true upwind SBP operators. Future work could thus consider further suitable combinations of first- and second-derivative upwind SBP operators with a focus on the specific form of the global SBP property which includes the influence of artificial diffusion introduced by the various types of numerical fluxes. For the prototype non-linear testcase of the viscous Burgers' equation, the numerical results indicate an analogous behavior concerning grid-independent stability of IMEX time integration depending on the evaluated diffusion fluxes. This performance should also be validated by theoretical means. Furthermore, an extension of this investigation to the multi-dimensional case as hinted at the end of Section 3.3 would be desirable.

A second contribution of this work in the context of advanced time integration concerns the Patankar approach to construct unconditionally positive and conservative numerical schemes for ordinary differential equations in production-destruction form. Hereby, we considered the behavior of these schemes for production-destruction formulations resulting from semi-discretized partial differential equations. More specifically, this involved the study of truncation errors of Patankar-modified explicit Runge-Kutta schemes for the classical examples of the linear advection equation and the linear heat equation discretized by low order finite difference schemes in conservation form. In fact, a first instructive example showed that the first order non-conservative Patankar-Euler method naively applied to the semi-discrete linear heat equation results in an inconsistent fully discrete method. For the conservative mPaRK2 scheme, which is second order accurate for ordinary differential equations, no order reduction occurs if the exact solution is sufficiently smooth and is bounded by a strictly positive lower limit. However, in the case that the exact solution vanishes at discrete points, order reduction effectively occurs as confirmed by the corresponding numerical results. A possible improvement of the original mPaRK2 scheme was given by a related approach based on a direct Patankar-type correction of the explicit part within the implicit trapezoidal rule. Since the enforcement of positivity also restricts the time step sizes of classical implicit methods with the exception of first order schemes, the Patankar trick was then applied to a third-order accurate implicit SDIRK scheme in order to combine unconditional positivity with higher order in time.

Subsequently, the resulting MPSDIRK3 scheme was applied to the time integration of the DG-discretized shallow water equations on unstructured triangular grids – aiming at a robust and accurate simulation of wetting and drying processes. For this purpose, a suitable production-destruction equation for the cell averages of water height was extracted from the DG-discretized continuity equation. The weights introduced by the Patankar scheme are thereby designed to sufficiently reduce the outgoing water fluxes constituting the destruction terms. Applying the corresponding weights to the ingoing fluxes which constitute the production terms then recovers mass conservation. The MPSDIRK3 scheme is third order accurate away from the wet/dry front but we can expect at most first order accuracy of the time integration scheme in wet/dry transition areas. Nonetheless, this matches with the approximation order in space, since at the shoreline, dissipation mechanisms such as modal filtering locally reduce the accuracy of the spatial discretization to first order as well. Numerical experiments were carried out for several classical test cases for shallow water flows regarding wetting and drying and well-balancedness in order to study the performance and accuracy of the implicit MPSDIRK3 time integration in comparison to the explicit TVD-RK3 scheme. Disposing of the restriction on the time step size for positivity preservation, the DG scheme with MPSDIRK3 time integration may employ larger time steps compared to explicit TVD-RK time integration and needs less computational time on stiff grids.

This thesis also contains a review on numerical methods for wetting and drying shallow water flows showing that the variety of different wetting and drying treatments present in current algorithms is impressive. Most numerical schemes try to satisfy the key properties of positivity preservation, local and global mass conservation, well-balancedness, non-permeability of dry areas, and elimination of artificial pressure gradients. For first order schemes in space and time, wetting and drying is managed more easily because many of the desired properties can already be fulfilled by a suitable choice of the numerical flux function and the source term discretization. For higher order schemes such as the DG scheme all of these properties require more effort. For positivity preservation, this specifically comprises guaranteeing positivity of the water height at additional interior nodes, employing SSP time integration with a specific time step restriction, and preventing artificial pressure gradients. With respect to the efficiency of implicit time integration, there still is room for further development. In some cases, difficulties may arise in form of a slow convergence of the Newton solver due to non-smooth switches caused by the wetting and drying treatment. For instance, these switches include the computation of the velocity from the discharge, which is unstable for small water height, as well as some computations within the Riemann solver. The check for neglecting gravitational forces usually is a non-smooth switch as well. In this context, a promising approach by Le et al. [106] is to regulate the gravitational forces by a blending parameter in partially dry cells to enable fast Newton convergence of the implicit solver.

# Bibliography

- [1] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. New York: Dover Publications, Inc., 1974.
- [2] M. A. Alhawwary and Z. J. Wang. “A study of DG methods for diffusion using the combined-mode analysis”. In: AIAA 2019-1157. 2019.
- [3] Y. Allaneau and A. Jameson. “Connections between the filtered discontinuous Galerkin method and the flux reconstruction approach to high order discretizations”. In: *Computer Methods in Applied Mechanics and Engineering* 200.49 (2011), pp. 3628–3636.
- [4] Y. Allaneau and A. Jameson. “Kinetic energy conserving discontinuous Galerkin scheme”. In: *Proceedings of the 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. AIAA-2011-198. 2011.
- [5] D. Arnold. “An interior penalty finite element method with discontinuous elements”. In: *SIAM J. Numer. Anal.* 19 (1982), pp. 742–760.
- [6] D. Arnold, F. Brezzi, B. Cockburn, and L. Marini. “Unified analysis of discontinuous Galerkin methods for elliptic problems”. In: *SIAM J. Numer. Anal.* 39 (2002), pp. 1749–1779.
- [7] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. “Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations”. In: *Appl. Numer. Math.* 25.2 (1997), pp. 151–167.
- [8] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton. “Implicit-explicit methods for time-dependent partial differential equations”. In: *SIAM J. Numer. Anal.* 32.3 (1995), pp. 797–823.
- [9] K. Asthana and A. Jameson. “High-order flux reconstruction schemes with minimal dispersion and dissipation”. In: *Journal of Scientific Computing* 62.3 (2015), pp. 913–944.
- [10] E. Audusse, F. Bouchut, M.-O. Bristeau, R. Klein, and B. Perthame. “A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows”. In: *SIAM J. Sci. Comput.* 25 (2004), pp. 2050–2065.
- [11] A. Balzano. “Evaluation of methods for numerical simulation of wetting and drying in shallow water flow models”. In: *Coastal Engineering* 34 (1998), pp. 83–107.

- [12] F. Bassi, N. Franchina, A. Ghidoni, and S. Rebay. “A numerical investigation of a spectral-type nodal collocation discontinuous Galerkin approximation of the Euler and Navier-Stokes equations”. In: *International Journal for Numerical Methods in Fluids* 71.10 (2013), pp. 1322–1339.
- [13] F. Bassi and S. Rebay. “A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations”. In: *J. Comput. Phys.* 131 (1997), pp. 267–279.
- [14] F. Bassi and S. Rebay. “GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations”. In: *Discontinuous Galerkin Methods*. Ed. by B. Cockburn, G. E. Karniadakis, and C.-W. Shu. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 197–208.
- [15] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, and M. Savini. “A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows”. In: *Proceedings of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics*. Technologisch Instituut, Antwerpen, Belgium. 1997, pp. 99–109.
- [16] P. D. Bates and J.-M. Hervouet. “A new method for moving-boundary hydrodynamic problems in shallow water”. In: *Proceedings of the Royal Society of London, Series A* 455 (1999), pp. 3107–3128.
- [17] C. E. Baumann and J. T. Oden. “A discontinuous hp finite element method for convection-diffusion problems”. In: *Comput. Methods Appl. Mech. Eng.* 175 (1999), pp. 311–341.
- [18] N. Beisiegel and J. Behrens. “Quasi-nodal third-order Bernstein polynomials in a discontinuous Galerkin model for flooding and drying”. In: *Environ. Earth Sci.* 74 (2015), pp. 7275–7284.
- [19] F. J. Blom. “A monolithical fluid-structure interaction algorithm applied to the piston problem”. In: *Comput. Methods Appl. Mech. Engrg.* 167 (1998), pp. 369–391.
- [20] M. G. Blyth and C. Pozrikidis. “A comparison of interpolation grids over the triangle or the tetrahedron”. In: *J. Eng. Math.* 56 (2006), pp. 263–272.
- [21] O. Bokhove. “Flooding and drying in discontinuous Galerkin finite-element discretizations of shallow-water equations. Part 1: one dimension”. In: *J. Sci. Comput.* 22 (2005), pp. 47–82.
- [22] A. Bollermann, G. Chen, A. Kurganov, and S. Noelle. “A well-balanced reconstruction of wet/dry fronts for the shallow water equations”. In: *J. Sci. Comput.* 56 (2013), pp. 267–290.
- [23] A. Bollermann, S. Noelle, and M. Lukáčová-Medvidová. “Finite volume evolution Galerkin methods for the shallow water equations with dry beds”. In: *Commun. Comput. Phys.* 10 (2011), pp. 371–404.
- [24] C. Bolley and M. Crouzeix. “Conservation de la positivité lors de la discréétisation des problèmes d’ évolution paraboliques”. In: *RAIRO Anal. Numér.* 12 (1978), pp. 237–245.

- [25] S. Bremicker-Trübelhorn and S. Ortlob. “On multirate GARK schemes with adaptive micro step sizes for fluid-structure interaction: order conditions and preservation of the geometric conservation law”. In: *Aerospace* 4.1 (2017).
- [26] F. Brezzi, G. Manzini, D. Marini, P. Pietra, and A. Russo. “Discontinuous finite elements for diffusion problems”. In: *Atti Convegno in onore di F. Brioschi (Milano 1997), Istituto Lombardo, Accademia di Scienze e Lettere.* 1999, pp. 197–217.
- [27] S. Bryson, Y. Epshteyn, A. Kurganov, and G. Petrova. “Central-upwind scheme on triangular grids for the Saint-Venant system of shallow water equations”. In: *AIP Conf. Proc.* 1389 (2011), pp. 686–689.
- [28] S. Bunya, E. J. Kubatko, J. J. Westerling, and C. Dawson. “A wetting and drying treatment for the Runge-Kutta discontinuous Galerkin solution to the shallow water equations”. In: *Comput. Methods Appl. Mech. Engrg.* 198 (2009), pp. 1548–1562.
- [29] H. Burchard, E. Deleersnijder, and A. Meister. “A high-order conservative Patankar-type discretisation for stiff systems of production-destruction equations”. In: *Appl. Numer. Math.* 47 (2003), pp. 1–30.
- [30] M. P. Calvo, J. de Frutos, and J. Novo. “Linearly implicit Runge-Kutta methods for advection-reaction-diffusion equations”. In: *Appl. Numer. Math.* 37.4 (2001), pp. 535–549.
- [31] A. S. Candy. “An implicit wetting and drying approach for non-hydrostatic baroclinic flows in high aspect ratio domains”. In: *Adv. Water Resour.* 102 (2017), pp. 188–205.
- [32] M. H. Carpenter, D. Gottlieb, and S. Abarbanel. “Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes”. In: *J. Comput. Phys.* 111.2 (1994), pp. 220–236.
- [33] M. H. Carpenter, J. Nordström, and D. Gottlieb. “A stable and conservative interface treatment of arbitrary spatial accuracy”. In: *J. Comput. Phys.* 148 (1999), pp. 341–365.
- [34] J. R. Cash. “Diagonally implicit Runge-Kutta formulae with error estimates”. In: *IMA Journal of Applied Mathematics* 24 (1979), pp. 293–301.
- [35] N. Castel, G. Cohen, and M. Duruflé. “Application of discontinuous Galerkin spectral method on hexahedral elements for aeroacoustic”. In: *Journal of Computational Acoustics* 17.02 (2009), pp. 175–196.
- [36] P. Castonguay, P. E. Vincent, and A. Jameson. “A new class of high-order energy stable flux reconstruction schemes for triangular elements”. In: *Journal of Scientific Computing* 51.1 (2012), pp. 224–256.
- [37] P. Castonguay, D.M. Williams, P.E. Vincent, and A. Jameson. “Energy stable flux reconstruction schemes for advection-diffusion problems”. In: *Computer Methods in Applied Mechanics and Engineering* 267 (2013), pp. 400–417.
- [38] V. Casulli. “A high-resolution wetting and drying algorithm for free-surface hydrodynamics”. In: *Int. J. Numer. Meth. Fluids* 60 (2009), pp. 391–408.
- [39] P. Chandrashekar. “Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations”. In: *Communications in Computational Physics* 14 (05 2013), pp. 1252–1286.

- [40] Y. Cheng and C.-W. Shu. “Superconvergence of local discontinuous Galerkin methods for one-dimensional convection-diffusion equations”. In: *Computers & Structures* 87.11 (2009), pp. 630–641.
- [41] S. Clain, C. Reis, R. Costa, J. Figueiredo, M. A. Baptista, and J. M. Miranda. “Second-order finite volume with hydrostatic reconstruction for tsunami simulation”. In: *J. Adv. Model. Earth Syst.* 8 (2016), pp. 1691–1713.
- [42] B. Cockburn, S. Hou, and C.-W. Shu. “TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case”. In: *Math. Comp.* 54 (1990), pp. 545–581.
- [43] B. Cockburn and C.-W. Shu. “Runge-Kutta discontinuous Galerkin methods for convection-dominated problems”. In: *J. Sci. Comp.* 16 (2001), pp. 173–261.
- [44] B. Cockburn and C.-W. Shu. “The local discontinuous Galerkin method for time-dependent convection-diffusion systems”. In: *SIAM J. Numer. Anal.* 35.6 (1998), pp. 2440–2463.
- [45] G. Cohen, X. Ferrieres, and S. Pernet. “A spatial high-order hexahedral discontinuous Galerkin method to solve Maxwell’s equations in time domain”. In: *Journal of Computational Physics* 217.2 (2006), pp. 340–363.
- [46] E. M. Constantinescu and A. Sandu. “Multirate timestepping methods for hyperbolic conservation laws”. In: *J. Sci. Comput.* 33 (2007), pp. 239–278.
- [47] D. C. Del Rey Fernández, P. D. Boom, and D. W. Zingg. “A generalized framework for nodal first derivative summation-by-parts operators”. In: *J. Comput. Phys.* 266 (2014), pp. 214–239.
- [48] D. C. Del Rey Fernández, J. Hicken, and D. W. Zingg. “Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations”. In: *Computers & Fluids* 95 (2014), pp. 171–196.
- [49] D. C. Del Rey Fernández and D. W. Zingg. “Generalized summation-by-parts operators for the second derivative”. In: *SIAM Journal on Scientific Computing* 37.6 (2015), A2840–A2864.
- [50] Ph. Delorme, P. Mazet, C. Peyret, and Y. Ventribout. “Computational aeroacoustics applications based on a discontinuous Galerkin method”. In: *Comptes Rendus Mécanique* 333.9 (2005), pp. 676–682.
- [51] M. Dubiner. “Spectral methods on triangles and other domains”. In: *J. of Scientific Computing* 6 (1991), pp. 345–390.
- [52] A. Duran, Q. Liang, and F. Marche. “On the well-balanced numerical discretization of shallow water equations on unstructured meshes”. In: *J. Comput. Phys.* 235 (2013), pp. 565–586.
- [53] A. Duran and F. Marche. “Recent advances on the discontinuous Galerkin method for shallow water equations with topography source terms”. In: *Computers & Fluids* 101 (2014), pp. 88–104.
- [54] A. Ern, S. Piperno, and K. Djadeli. “A well-balanced Runge-Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying”. In: *Int. J. Numer. Meth. Fluids* 58 (2008), pp. 1–25.

- [55] S. Fagherazzi, D. J. Furbish, P. Rasetarinera, and M. Y. Hussaini. “Application of the discontinuous spectral Galerkin method to groundwater flow”. In: *Advances in Water Resources* 27.2 (2004), pp. 129–140.
- [56] T. C. Fisher, M. H. Carpenter, J. Nordström, N. K. Yamaleev, and C. Swanson. “Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions”. In: *J. Comput. Phys.* 234 (2013), pp. 353–375.
- [57] U. S. Fjordholm, S. Mishra, and E. Tadmor. “Well-balanced and energy stable schemes for the shallow water equations with discontinuous topography”. In: *Journal of Computational Physics* 230 (2011), pp. 5587–5609.
- [58] B. M. Froehle and P.-O. Persson. “A high-order implicit-explicit fluid-structure interaction method for flapping flight”. In: *21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences*. AIAA 2013-2690. San Diego, 2013.
- [59] G. Fu and C.-W. Shu. “Analysis of an embedded discontinuous Galerkin method with implicit-explicit time-marching for convection-diffusion problems”. In: *Int. J. Numer. Anal. Model.* 14 (2017), pp. 477–499.
- [60] J. M. Gallardo, C. Parés, and M. Castro. “On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas”. In: *J. Comput. Phys.* 227 (2007), pp. 574–601.
- [61] G. J. Gassner. “A kinetic energy preserving nodal discontinuous Galerkin spectral element method”. In: *Int. J. Numer. Meth. Fluids* 76 (2014), pp. 28–50.
- [62] G. J. Gassner. “A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods”. In: *SIAM J. Sci. Comput.* 35 (2013), A1233–A1253.
- [63] G. J. Gassner and A. D. Beck. “On the accuracy of high-order discretizations for underresolved turbulence simulations”. In: *Theor. Comp. Fluid Dyn.* 27 (2013), pp. 221–237.
- [64] G. J. Gassner, F. Hindenlang, and C.-D. Munz. “A Runge-Kutta based discontinuous Galerkin method with time accurate local time stepping”. In: *Adaptive High-Order Methods in Computational Fluid Dynamics*. Ed. by Z. J. Wang. World Scientific, 2011, pp. 95–118.
- [65] G. J. Gassner and D. A. Kopriva. “A comparison of the dispersion and dissipation errors of Gauss and Gauss-Lobatto discontinuous Galerkin spectral element methods”. In: *SIAM J. Sci. Comput.* 33.5 (2011), pp. 2560–2579.
- [66] G. J. Gassner, F. Lörcher, and C.-D. Munz. “A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes”. In: *Journal of Computational Physics* 224 (2007), pp. 1049–1063.
- [67] G. J. Gassner, A. R. Winters, F. J. Hindenlang, and D. A. Kopriva. “The BR1 scheme is stable for the compressible Navier-Stokes equations”. In: *J. Sci. Comput.* 77 (2018), pp. 154–200.

- [68] G. J. Gassner, A. R. Winters, and D. A Kopriva. “A well-balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations”. In: *Appl. Math. Comput.* 272 (2016), pp. 291–308.
- [69] G. J. Gassner, A. R. Winters, and D. A. Kopriva. “Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations”. In: *Journal of Computational Physics* 327 (2016), pp. 39–66.
- [70] G. J. Gassner, A. R. Winters, and D. A. Kopriva. “Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations”. In: *Journal of Computational Physics* 327 (2016), pp. 39–66.
- [71] F. X. Giraldo, J. S. Hesthaven, and T. Warburton. “Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations”. In: *Journal of Computational Physics* 181.2 (2002), pp. 499–525.
- [72] S. Gottlieb, C.-W. Shu, and E. Tadmor. “Strong stability-preserving high-order time discretization methods”. In: *SIAM Review* 43 (2001), pp. 89–112.
- [73] O. Gourge, R. Comblen, J. Lambrechts, T. Kärnä, V. Legat, and E. Deleersnijder. “A flux-limiting wetting-drying method for finite-element shallow-water models, with application to the Scheldt Estuary”. In: *Adv. Water Resour.* 32 (2009), pp. 1726–1739.
- [74] W. Guo, X. Zhong, and J.M. Qiu. “Superconvergence of discontinuous Galerkin and local discontinuous Galerkin methods: Eigen-structure analysis based on Fourier approach”. In: *Journal of Computational Physics* 235 (2013), pp. 458–485.
- [75] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations I. Nonstiff problems*. Springer-Verlag, 1993.
- [76] A. Harten, P. D. Lax, and B. van Leer. “On upstream differencing and Godunov-type schemes for hyperbolic conservation laws”. In: *SIAM Review* 25 (1983), pp. 35–61.
- [77] J. Hicken, D. C. Del Rey Fernández, and D. W. Zingg. “Multidimensional summation-by-parts operators: general theory and application to simplex elements”. In: *SIAM Journal on Scientific Computing* 38 (2016), A1935–A1958.
- [78] I. Higueras. “Representations of Runge-Kutta methods and strong stability preserving methods”. In: *SIAM J. Numer. Anal.* 43 (2005), pp. 924–948.
- [79] B. van’t Hof and A. E. P. Veldman. “Mass, momentum and energy conserving (MaMEC) discretizations on general grids for the compressible Euler and shallow water equations”. In: *J. Comput. Phys.* 231 (2012), pp. 4723–4744.
- [80] Z. Horváth. “Positivity of Runge-Kutta and diagonally split Runge-Kutta methods”. In: *Appl. Numer. Math.* 28 (1998), pp. 309–326.
- [81] Z. Horváth, J. Waser, R. A. P. Perdigão, A. Konev, and G. Böschl. “A two-dimensional numerical scheme of dry/wet fronts for the Saint-Venant system of shallow water equations”. In: *Int. J. Numer. Meth. Fluids* 77 (2014), pp. 159–182.
- [82] F. Q. Hu, M. Y. Hussaini, and P. Rasetarinera. “An analysis of the discontinuous Galerkin method for wave propagation problems”. In: *Journal of Computational Physics* 151.2 (1999), pp. 921–946.

- [83] W. Hundsdorfer, A. Mozartova, and V. Savcenco. *Analysis of explicit multirate and partitioned Runge-Kutta schemes for conservation laws*. Tech. rep. MAS-E0715. CWI Amsterdam, 2007.
- [84] W. Hundsdorfer and J. G. Verwer. *Numerical solution of time-dependent advection-diffusion-reaction equations*. Springer Series in Computational Mathematics. Springer-Verlag Berlin Heidelberg, 2003.
- [85] H. T. Huynh. “A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods”. In: AIAA 2007-4079. 2007.
- [86] H. T. Huynh. “A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion”. In: AIAA 2009-403. 2009.
- [87] H. T. Huynh, Z. J. Wang, and P. E. Vincent. “High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids”. In: *Computers & Fluids* 98 (2014), pp. 209–220.
- [88] K. Ishiko, N. Ohnishi, K. Ueno, and K. Sawada. “Implicit large eddy simulation of two-dimensional homogeneous turbulence using weighted compact nonlinear scheme”. In: *J. Fluids Eng.* 131 (6 2009), 061401:1–061401:14.
- [89] F. Ismail and P. L. Roe. “Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks”. In: *J. Comput. Phys.* 228 (2009), pp. 5410–5436.
- [90] Z. Jackiewicz and R. Vermiglio. “Order conditions for partitioned Runge-Kutta methods”. In: *Applications of Mathematics* 45 (2000), pp. 301–316.
- [91] A. Jameson. “A proof of the stability of the spectral difference method for all orders of accuracy”. In: *J. Sci. Comput.* 45 (2010), pp. 348–358.
- [92] A. Jameson. “Formulation of kinetic energy preserving conservative schemes for gas dynamics and direct numerical simulation of one-dimensional viscous compressible flow in a shock tube using entropy and kinetic energy preserving schemes”. English. In: *J. Sci. Comput.* 34 (2008), pp. 188–208.
- [93] A. Javadi, M. Pasandideh-Fard, and M. Malek-Jafarian. “Modification of k- $\epsilon$  turbulent model using kinetic energy-preserving method”. In: *Numerical Heat Transfer, Part B: Fundamentals* 68 (2015), pp. 554–577.
- [94] H. M. Kalita and A. K. Sarma. “An implicit scheme for shallow water flow with wet dry interface”. In: *Water Resour.* 45 (2018), pp. 61–68.
- [95] A. Kanevsky, M. H. Carpenter, D. Gottlieb, and J. S. Hesthaven. “Application of implicit-explicit high order Runge-Kutta methods to discontinuous-Galerkin schemes”. In: *Journal of Computational Physics* 225.2 (2007), pp. 1753–1781.
- [96] T. Kärnä, B. de Brye, O. Gourgue, J. Lambrechts, R. Combлен, V. Legat, and E. Deleersnijder. “A fully implicit wetting-drying method for DG-FEM shallow water models, with an application to the Scheldt Estuary”. In: *Comp. Meth. Appl. Mech. Engng.* 200 (2011), pp. 509–524.
- [97] G. Karniadakis and S. Sherwin. *Spectral/hp element methods for computational fluid dynamics*. 2nd ed. Oxford University Press, 2005.

- [98] S. Kopecz and A. Meister. “On order conditions for modified Patankar-Runge-Kutta schemes”. In: *Applied Numerical Mathematics* 123 (2018), pp. 159–179.
- [99] S. Kopecz and A. Meister. “On the existence of three-stage third-order modified Patankar-Runge-Kutta schemes”. In: *Numerical Algorithms* 81 (2019), pp. 1473–1484.
- [100] S. Kopecz and A. Meister. “Unconditionally positive and conservative third order modified Patankar-Runge-Kutta discretizations of production-destruction systems”. In: *BIT Numerical Mathematics* 58 (2018), pp. 691–728.
- [101] D. A. Kopriva and G. J. Gassner. “On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods”. English. In: *J. Sci. Comput.* 44 (2010), pp. 136–155.
- [102] D. A. Kopriva and John H. Kolias. “A conservative staggered-grid Chebyshev multidomain method for compressible flows”. In: *Journal of Computational Physics* 125 (1996), pp. 244–261.
- [103] D. A. Kopriva, S. L. Woodruff, and M. Y. Hussaini. “Computation of electromagnetic scattering with a non-conforming discontinuous spectral element method”. In: *International Journal for Numerical Methods in Engineering* 53.1 (2002), pp. 105–122.
- [104] H.-O. Kreiss and G. Scherer. “Finite element and finite difference methods for hyperbolic partial differential equations”. In: *Mathematical Aspects of Finite Elements in Partial Differential Equations*. Ed. by Carl de Boor. Academic Press, 1974, pp. 195–212.
- [105] A. Kurganov and G. Petrova. “A second-order well-balanced positivity preserving central-upwind scheme for the Saint-Venant system”. In: *Commun. Math. Sci.* 5 (2007), pp. 133–160.
- [106] H.-A. Le, J. Lambrechts, S. Ortleb, N. Gratiot, E. Deleersnijder, and S. Soares-Frazão. “An implicit wetting-drying algorithm for the discontinuous Galerkin method: application to the Tonle Sap, Mekong River Basin”. In: *Environ. Fluid. Mech.* 20 (2020), pp. 923–951.
- [107] H. Lee and N. Lee. “Wet-dry moving boundary treatment for Runge-Kutta discontinuous Galerkin shallow water equation model”. In: *KSCE J. Civ. Eng.* 20 (2016), pp. 978–989.
- [108] B. van Leer. “Flux-vector splitting for the Euler equations”. English. In: *Eighth International Conference on Numerical Methods in Fluid Dynamics*. Ed. by E. Krause. Vol. 170. Lecture Notes in Physics. Springer Berlin Heidelberg, 1982, pp. 507–512.
- [109] B. van Leer, M. Lo, R. Gitik, and S. Nomura. “A venerable family of discontinuous Galerkin schemes for diffusion revisited”. In: *Adaptive High-Order Methods in Computational Fluid Dynamics*. Ed. by Z. J. Wang. World Scientific, 2011, pp. 185–201.
- [110] B. van Leer and M. van Raalte M. Lo. “A discontinuous Galerkin method for diffusion based on recovery”. In: AIAA 2007-4083. 2007.
- [111] B. van Leer and S. Nomura. “Discontinuous Galerkin for diffusion”. In: *17th AIAA Computational Fluid Dynamics Conference*. AIAA-2005-5108. 2005.

- [112] E. Lefrançois and J.-P. Boufflet. “An introduction to fluid-structure interaction: application to the piston problem”. In: *SIAM Rev.* 52.4 (2010), pp. 747–767.
- [113] R. J. LeVeque. “Balancing source terms and flux gradients on high-resolution Godunov methods: the quasi-steady wave-propagation algorithm”. In: *J. Comput. Phys.* 146 (1998), pp. 346–365.
- [114] Q. Liang and F. Marche. “Numerical resolution of well-balanced shallow water equations with complex source terms”. In: *Adv. Water Resour.* 32 (2009), pp. 873–884.
- [115] K. Lipnikov, G. Manzini, and M. Shashkov. “Mimetic finite difference method”. In: *Journal of Computational Physics* 257 (2014), pp. 1163–1227.
- [116] H. Liu and J. Yan. “The direct discontinuous Galerkin (DDG) method for diffusion with interface corrections”. In: *Communications in Computational Physics* 8 (2010), pp. 541–564.
- [117] H. Liu and J. Yan. “The direct discontinuous Galerkin (DDG) methods for diffusion problems”. In: *SIAM Journal on Numerical Analysis* 47 (2009), pp. 675–698.
- [118] Y. Liu, M. Vinokur, and Z. J. Wang. “Spectral difference method for unstructured grids I: Basic formulation”. In: *Journal of Computational Physics* 216 (2006), pp. 780–801.
- [119] M. Lo and B. van Leer. “Analysis and implementation of recovery-based discontinuous Galerkin for diffusion”. In: *19th AIAA Computational Fluid Dynamics Conference*. AIAA-2009-3786. 2009.
- [120] H. Luo, L. Luo, R. Nourgaliev, V. A. Mousseau, and N. Dinh. “A reconstructed discontinuous Galerkin method for the compressible Navier-Stokes equations on arbitrary grids”. In: *J. Comput. Phys.* 229 (2010), pp. 6961–6978.
- [121] C. B. Macdonald, S. Gottlieb, and S. J. Ruuth. “A numerical study of diagonally split Runge-Kutta methods for PDEs with discontinuities”. In: *J. Sci. Comput.* 36 (2008), pp. 89–112.
- [122] J. Manzanero, E. Ferrer, G. Rubio, and E. Valero. “On the role of numerical dissipation in stabilising under-resolved turbulent simulations using discontinuous Galerkin methods”. In: (2018). arXiv preprint arXiv:1805.10519.
- [123] J. Manzanero, G. Rubio, E. Ferrer, and E. Valero. “Dispersion-dissipation analysis for advection problems with nonconstant coefficients: applications to discontinuous Galerkin formulations”. In: *SIAM Journal on Scientific Computing* 40.2 (2018), A747–A768.
- [124] F. Marche, P. Bonneton, P. Fabrie, and N. Seguin. “Evaluation of well-balanced bore-capturing schemes for 2D wetting and drying processes”. In: *Int. J. Numer. Meth. Fluids* 53 (2017), pp. 867–894.
- [125] S. Marras, M. A. Kopera, E. M. Constantinescu, J. Suckale, and F. X. Giraldo. *A continuous/discontinuous Galerkin solution of the shallow water equations with dynamic viscosity, high-order wetting and drying, and implicit time integration*. 2016. arXiv: 1607.04547.

- [126] K. Mattsson. “Diagonal-norm upwind SBP operators”. In: *Journal of Computational Physics* 335 (2017), pp. 283–310.
- [127] K. Mattsson, M. Svärd, and J. Nordström. “Stable and accurate artificial dissipation”. In: *J. Sci. Comput.* 21 (2004), pp. 57–79.
- [128] K. Mattsson, M. Svärd, and M. Shoeybi. “Stable and accurate schemes for the compressible Navier-Stokes equations”. In: *Journal of Computational Physics* 227.4 (2008), pp. 2293–2316.
- [129] D. J. Mavriplis and Z. Yang. “Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes”. In: *J. Comput. Phys.* 213 (2006), pp. 557–573.
- [130] S. C. Medeiros and S. C. Hagen. “Review of wetting and drying algorithms for numerical tidal flow models”. In: *Int. J. Numer. Meth. Fluids* 71 (2013), pp. 473–487.
- [131] A. Meister and S. Ortlob. “A positivity preserving and well-balanced DG scheme using finite volume subcells in almost dry regions”. In: *Applied Mathematics and Computation* 272 (2016), pp. 259–273.
- [132] A. Meister and S. Ortlob. “On unconditionally positive implicit time integration for the DG scheme applied to shallow water flows”. In: *International Journal for Numerical Methods in Fluids* 76.2 (2014), pp. 69–94.
- [133] A. Meister and S. Ortlob. “The DG Scheme on triangular grids with adaptive modal and variational filtering routines applied to shallow water flows”. In: *Recent Developments in the Numerics of Nonlinear Hyperbolic Conservation Laws: Lectures Presented at a Workshop at the Mathematical Research Institute Oberwolfach, Germany, Jan 15 – 21, 2012*. Ed. by R. Ansorge, H. Bijl, A. Meister, and Th. Sonar. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 253–266.
- [134] A. Meister, S. Ortlob, and Th. Sonar. “Application of spectral filtering to discontinuous Galerkin methods on triangulations”. In: *Numer. Methods Partial Differ. Equ.* 28 (2012), pp. 1840–1868.
- [135] A. Meister, S. Ortlob, and Th. Sonar. “New adaptive modal and DTV filtering routines for the DG method on triangular grids applied to the Euler equations”. In: *Int. J. Geomath.* 3 (2012), pp. 17–50.
- [136] Y. Morinishi. “Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-Mach number flows”. In: *J. Comput. Phys.* 229 (2010), pp. 276–300.
- [137] Y. Morinishi, T. S. Lund, O. V. Vasilyev, and P. Moin. “Fully conservative higher order finite difference schemes for incompressible flow”. In: *J. Comput. Phys.* 143 (1998), pp. 90–124.
- [138] R. C. Moura, S. J. Sherwin, and J. Peiró. “Eigensolution analysis of spectral/hp continuous Galerkin approximations to advection-diffusion problems: Insights into spectral vanishing viscosity”. In: *Journal of Computational Physics* 307 (2016), pp. 401–422.
- [139] R. C. Moura, S. J. Sherwin, and J. Peiró. “Linear dispersion-diffusion analysis and its application to under-resolved turbulence simulations using discontinuous Galerkin spectral/hp methods”. In: *Journal of Computational Physics* 298 (2015), pp. 695–710.

- [140] N. C. Nguyen, J. Peraire, and B. Cockburn. “Hybridizable discontinuous Galerkin methods”. In: *Spectral and High Order Methods for Partial Differential Equations*. Ed. by Jan S. Hesthaven and Einar M. Rønquist. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 63–84.
- [141] C. Nielsen and C. Apelt. “Parameters affecting the performance of wetting and drying in a two-dimensional finite element long wave hydrodynamic model”. In: *J. Hydraul. Eng.* 129 (2003), pp. 628–636.
- [142] S. Noelle, Y. Xing, and C.-W. Shu. “High-order well-balanced finite volume WENO schemes for shallow water equation with moving water”. In: *Journal of Computational Physics* 226 (2007), pp. 29–58.
- [143] J. Nordström, K. Forsberg, C. Adamsson, and P. Eliasson. “Finite volume methods, unstructured meshes and strict stability for hyperbolic problems”. In: *Applied Numerical Mathematics* 45 (2003), pp. 453–473.
- [144] J. Nordström, J. Gong, E. van der Weide, and M. Svärd. “A stable and conservative high order multi-block method for the compressible Navier-Stokes equations”. In: *Journal of Computational Physics* 228.24 (2009), pp. 9020–9035.
- [145] S. Nüßlein, H. Ranocha, and D. I. Ketcheson. *Positivity-preserving adaptive Runge-Kutta methods*. 2020. arXiv: 2005.06268.
- [146] P. Olsson. “Summation by parts, projections, and stability. I”. In: *Mathematics of Computation* 64.211 (1995), pp. 1035–1065.
- [147] P. Olsson. “Summation by parts, projections, and stability. II”. In: *Mathematics of Computation* 64.212 (1995), pp. 1473–1493.
- [148] S. Ortlob. “A comparative Fourier analysis of discontinuous Galerkin schemes for advection-diffusion with respect to BR1, BR2, and local discontinuous Galerkin diffusion discretization”. In: *Mathematical Methods in the Applied Sciences* 43.13 (2020), pp. 7841–7863.
- [149] S. Ortlob. “A Fourier-type analysis of the Gauss and Gauss-Lobatto P1-discontinuous Galerkin methods for the linear advection-diffusion equation”. In: *AIP Conference Proceedings* 2116 (2019), p. 340003.
- [150] S. Ortlob. “A kinetic energy preserving DG scheme based on Gauss-Legendre points”. In: *J. Sci. Comput.* 71 (2017), pp. 1135–1168.
- [151] S. Ortlob. “Fourier analysis of DG schemes for advection-diffusion”. In: *PAMM* 20 (2020). accepted.
- [152] S. Ortlob. “L2-stability analysis of IMEX- $(\sigma, \mu)$  DG schemes for linear advection-diffusion equations”. In: *Applied Numerical Mathematics* 147 (2020), pp. 43–65.
- [153] S. Ortlob and M. Franke. “On the benefit of the summation-by-parts property on interior nodal sets”. In: *ECCM - ECFD 2018 : Proceedings of the 6th European Conference on Computational Mechanics (Solids, Structures and Coupled Problems) and the 7th European Conference on Computational Fluid Dynamics*. Ed. by R. Owen, R. de Borst, J. Reese, and C. Pearce. 2018, pp. 77–88.

- [154] S. Osher and R. Sanders. “Numerical approximations to nonlinear conservation laws with locally varying time and space grids”. In: *Math. Comput.* 41 (1983), pp. 321–336.
- [155] S. V. Patankar. *Numerical heat transfer and fluid flow*. Series on Computational Methods in Mechanics and Thermal Science. Hemisphere Publishing Corporation (CRC Press, Taylor & Francis Group), 1980.
- [156] M. Pelanti, F. Bouchut, and A. Mangeney. “A Riemann solver for single-phase and two-phase shallow flow models based on relaxation. Relations with Roe and VF Roe solvers”. In: *J. Comput. Phys.* 230 (2011), pp. 515–550.
- [157] J. Peraire and P.-O. Persson. “The compact discontinuous Galerkin (CDG) method for elliptic problems”. In: *SIAM J. Sci. Comput.* 30 (2008), pp. 1806–1824.
- [158] P.-O. Persson and J. Peraire. “Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier–Stokes equations”. In: 30 (2008), pp. 2709–2733.
- [159] P.-O. Persson and J. Peraire. “Sub-cell shock capturing for discontinuous Galerkin methods”. In: *44th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA-2006-0112. 2006.
- [160] S. Piperno. “Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2D inviscid aeroelastic simulations”. In: *International Journal for Numerical Methods in Fluids* 25 (1997), pp. 1207–1226.
- [161] Serge Piperno. “Simulation numérique de phénomènes d’interaction fluide-structure”. PhD thesis. Ecole Nationale des Pont et Chaussées, 1995.
- [162] S. Quaegebeur, S. Nadarajah, F. Navah, and P. Zwanenburg. “Stability of energy stable flux reconstruction for the diffusion problem using compact numerical fluxes”. In: *SIAM Journal on Scientific Computing* 41.1 (2019), A643–A667.
- [163] M. H. van Raalte. “Multigrid analysis and embedded boundary conditions for discontinuous Galerkin discretization”. PhD thesis. University of Amsterdam, 2004.
- [164] G. Rainwater and M. Tokman. “A new class of split exponential propagation iterative methods of Runge-Kutta type (sEPIRK) for semilinear systems of ODEs”. In: *Journal of Computational Physics* 269 (2014), pp. 40–60.
- [165] H. Ranocha. “Shallow water equations: split-form, entropy stable, well-balanced, and positivity preserving numerical methods”. In: *Int. J. Geomath.* 8 (2017), pp. 85–133.
- [166] H. Ranocha, P. Öffner, and Th. Sonar. “Summation-by-parts operators for correction procedure via reconstruction”. In: *Journal of Computational Physics* 311 (2016), pp. 299–328.
- [167] M. Restelli and F. X. Giraldo. “A conservative discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in nonhydrostatic mesoscale modeling”. In: *SIAM Journal on Scientific Computing* 31.3 (2009), pp. 2231–2257.
- [168] M. Ricchiuto and A. Bollermann. “Stabilized residual distribution for shallow water simulations”. In: *J. Comput. Phys.* 228 (2009), pp. 1071–1115.
- [169] R. R. Rosales, B. Seibold, D. Shirokoff, and D. Zhou. “Unconditional stability for multistep ImEx schemes: Theory”. In: *SIAM Journal on Numerical Analysis* 55.5 (2017), pp. 2336–2360.

- [170] O. San and A. E. Staples. “High-order methods for decaying two-dimensional homogeneous isotropic turbulence”. In: *Computers & Fluids* 63 (2012), pp. 105–127.
- [171] A. Sandu and M. Günther. “A generalized-structure approach to additive Runge-Kutta methods”. In: *SIAM J. Numer. Anal.* 53 (2015), pp. 17–42.
- [172] M. Schlegel, O. Knoth, M. Arnold, and R. Wolke. “Multirate Runge-Kutta schemes for advection equations”. In: *Journal of Computational and Applied Mathematics* 226 (2009). Special Issue: Large scale scientific computations, pp. 345–357.
- [173] B. Seibold, D. Shirokoff, and D. Zhou. “Unconditional stability for multistep ImEx schemes: Practice”. In: *Journal of Computational Physics* 376 (2019), pp. 295–321.
- [174] B. Seny, J. Lambrechts, R. Combлен, V. Legat, and J. F. Remacle. “Multirate time stepping for accelerating explicit discontinuous Galerkin computations with application to geophysical flows”. In: *Int. J. Numer. Meth. Fluids* 71 (2012), pp. 41–64.
- [175] M. Shoeybi, M. Svärd, F. E. Ham, and P. Moin. “An adaptive implicit-explicit scheme for the DNS and LES of compressible flows on unstructured grids”. In: *Journal of Computational Physics* 229.17 (2010), pp. 5944–5965.
- [176] C.-W. Shu. “Total-variation diminishing time discretizations”. In: *SIAM J. Sci. Stat. Comp.* 9 (1988), pp. 1073–1084.
- [177] C.-W. Shu and S. Osher. “Efficient Implementation of essentially non-oscillatory shock capturing schemes II.” In: *J. Comput. Phys.* 83 (1989), pp. 32–78.
- [178] C.-W. Shu and S. Osher. “Efficient implementation of essentially non-oscillatory shock-capturing schemes”. In: *Journal of Computational Physics* 77 (1988), pp. 439–471.
- [179] M. Sonntag and C.-D. Munz. “Shock capturing for discontinuous Galerkin methods using finite volume subcells”. In: *Finite Volumes for Complex Applications VII-Elliptic, Parabolic and Hyperbolic Problems*. Ed. by J. Fuhrmann, M. Ohlberger, and C. Rohde. Cham: Springer International Publishing, 2014, pp. 945–953.
- [180] D. Stanescu, J. Xu, M.Y. Hussaini, and F. Farassat. “Computation of engine noise propagation and scattering off an aircraft”. In: *International Journal of Aeroacoustics* 1.4 (2002), pp. 403–420.
- [181] G. S. Stelling and S. P. A. Duinmeijer. “A staggered conservative scheme for every Froude number in rapidly varied shallow water flows”. In: *Int. J. Numer. Meth. Fluids* 43 (2003), pp. 1329–1354.
- [182] B. Strand. “Summation by parts for finite difference approximations for  $d/dx$ ”. In: *Journal of Computational Physics* 110.1 (1994), pp. 47–67.
- [183] V. Straub and S. Ortob. “Comparison of exponential-explicit (EXPEX) schemes in the domain-based implicit-explicit (IMEX) setting with IMEX-RK schemes for CFD applications”. In: *Proceedings of the 8th GACM Colloquium on Computational Mechanics for Young Scientists From Academia and Industry*. Ed. by T. Gleim and S. Lange. Kassel: kassel university press, 2019, pp. 303–306.
- [184] V. Straub, S. Ortob, P. Birken, and A. Meister. “A new domain-based implicit-explicit time stepping scheme based on the class of exponential integrators called sEPIRK”. In: *PAMM* 19.1 (2019), e201900142.

- [185] V. Straub, S. Ortlob, P. Birken, and A. Meister. “Adopting (s)EPIRK schemes in a domain-based IMEX setting”. In: *AIP Conference Proceedings* 1863.1 (2017), p. 410008.
- [186] P. K. Subbareddy and G. V. Candler. “A fully discrete, kinetic energy consistent finite-volume scheme for compressible flows”. In: *J. Comput. Phys.* 228 (2009), pp. 1347–1364.
- [187] M. Svärd and J. Nordström. “Review of summation-by-parts schemes for initial-boundary-value problems”. In: *J. Comput. Phys.* 268 (2014), pp. 17–38.
- [188] C. E. Synolakis. “The runup of solitary waves”. In: *J. Fluid. Mech.* 185 (1987), pp. 523–545.
- [189] E. Tadmor. “Numerical viscosity and the entropy condition for conservative difference schemes”. In: *Math. Comp.* 43 (1984), pp. 369–381.
- [190] E. Tadmor. “The numerical viscosity of entropy stable schemes for systems of conservation laws. I”. In: *Math. Comp.* 49 (1987), pp. 91–103.
- [191] H. Tang and G. Warnecke. “High resolution schemes for conservation laws and convection-diffusion equations with varying time and space grids”. In: *J. Comput. Math.* 24 (2006), pp. 121–140.
- [192] S. A. Teukolsky. “Short note on the mass matrix for Gauss-Lobatto grid points”. In: *J. Comput. Phys.* 283 (2015), pp. 408–413.
- [193] P. D. Thomas and C. K. Lombard. “Geometric conservation law and its application to flow computations on moving grids”. In: *AIAA Journal* 17 (1979), pp. 1030–1037.
- [194] M. Tokman. “A new class of exponential propagation iterative methods of Runge-Kutta type (EPIRK)”. In: *Journal of Computational Physics* 230.24 (2011), pp. 8762–8778.
- [195] M. Tokman and J. Loffeld. “Efficient design of exponential-Krylov integrators for large scale computing”. In: *Procedia Computer Science* 1.1 (2010), pp. 229–237.
- [196] K. Van den Abeele, C. Lacor, and Z.J. Wang. “On the stability and accuracy of the spectral difference method”. In: *Journal of Scientific Computing* 37.2 (2008), pp. 162–188.
- [197] S. Vater, N. Beisiegel, and J. Behrens. “A limiter-based well-balanced discontinuous Galerkin method for shallow-water flows with wetting and drying: One-dimensional case”. In: *Adv. Water Resour.* (2015), pp. 1–13.
- [198] B. C. Vermeire and S. Nadarajah. “Adaptive IMEX schemes for high-order unstructured methods”. In: *Journal of Computational Physics* 280 (2015), pp. 261–286.
- [199] B. C. Vermeire and P. E. Vincent. “On the behaviour of fully-discrete flux reconstruction schemes”. In: *Computer Methods in Applied Mechanics and Engineering* 315 (2017), pp. 1053–1079.
- [200] R. W. C. P. Verstappen and A. E. P. Veldman. “Symmetry-preserving discretization of turbulent flow”. In: *J. Comput. Phys.* 187 (2003), pp. 343–368.
- [201] J. G. Verwer, J. G. Blom, and W. Hundsdorfer. “An implicit-explicit approach for atmospheric transport-chemistry problems”. In: *Appl. Numer. Math.* 20.1 (1996), pp. 191–209.

- [202] P. E. Vincent, P. Castonguay, and A. Jameson. “A new class of high-order energy stable flux reconstruction schemes”. In: *Journal of Scientific Computing* 47.1 (2011), pp. 50–72.
- [203] P. E. Vincent, P. Castonguay, and A. Jameson. “Insights from von Neumann analysis of high-order flux reconstruction schemes”. In: *Journal of Computational Physics* 230.22 (2011), pp. 8134–8154.
- [204] H. Wang, C.-W. Shu, and Q. Zhang. “Stability analysis and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for nonlinear convection-diffusion problems”. In: *Appl. Math. Comput.* 272 (2016), pp. 237–258.
- [205] H. Wang, C.-W. Shu, and Q. Zhang. “Stability and error estimates of local discontinuous Galerkin methods with implicit-explicit time-marching for advection-diffusion problems”. In: *SIAM J. Numer. Anal.* 53.1 (2015), pp. 206–227.
- [206] Z. J. Wang and H. Gao. “A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids”. In: *Journal of Computational Physics* 228 (2009), pp. 8161–8186.
- [207] Z. J. Wang, L. Zhang, and Y. Liu. “Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional systems”. In: *Journal of Computational Physics* 194 (2004), pp. 716–741.
- [208] Wang, H., Wang, S., Zhang, Q., and Shu, C.-W. “Local discontinuous Galerkin methods with implicit-explicit time-marching for multi-dimensional convection-diffusion problems”. In: *ESAIM: M2AN* 50.4 (2016), pp. 1083–1105.
- [209] J. C. Warner, Z. Defne, K. Haas, and H. G. Arango. “A wetting and drying scheme for ROMS”. In: *Computers & Geosciences* (2013), pp. 54–61.
- [210] J. Watkins, K. Asthana, and A. Jameson. “A numerical analysis of the nodal Discontinuous Galerkin scheme via Flux Reconstruction for the advection-diffusion equation”. In: *Computers & Fluids* 139 (2016), pp. 233–247.
- [211] Y. Xing. “Exactly well-balanced discontinuous Galerkin methods for the shallow water equations with moving water equilibrium”. In: *Journal of Computational Physics* 257 (2014), pp. 536–553.
- [212] Y. Xing and C.-W. Shu. “A new approach of high order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms”. In: *Commun. Comput. Phys.* 1 (2006), pp. 100–134.
- [213] Y. Xing and X. Zhang. “Positivity-preserving well-balanced discontinuous Galerkin methods for the shallow water equations on unstructured triangular meshes”. In: *J. Sci. Comput.* 57 (2013), pp. 19–41.
- [214] Y. Xing, X. Zhang, and C.-W. Shu. “Positivity-preserving high order well-balanced discontinuous Galerkin methods for the shallow water equations”. In: *Adv. Water Resour.* 33 (2010), pp. 1476–1493.
- [215] J. Yu, C. Yan, and Z. Jiang. “On the use of the discontinuous Galerkin method for numerical simulation of two-dimensional compressible turbulence with shocks”. In: *Science China Physics, Mechanics & Astronomy* 57.9 (2014), pp. 1758–1770.

- [216] M. Zhang and C.-W. Shu. “An analysis of three different formulations of the discontinuous Galerkin method for diffusion equations”. In: *Mathematical Models and Methods in Applied Sciences* 13 (2003), pp. 395–413.
- [217] Q. Zhang and F. Gao. “A fully-discrete local discontinuous Galerkin method for convection-dominated Sobolev equation”. In: *J. Sci. Comput.* 51.1 (2012), pp. 107–134.
- [218] Q. Zhang and C.-W. Shu. “Stability analysis and a priori error estimates of the third order explicit Runge-Kutta discontinuous Galerkin method for scalar conservation laws”. In: *SIAM J. Numer. Anal.* 48.3 (2010), pp. 1038–1063.
- [219] X. Zhang and C.W. Shu. “On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes”. In: *Journal of Computational Physics* 229.23 (2010), pp. 8918–8934.
- [220] X. Zhang, Y. Xia, and C.-W. Shu. “Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes”. In: *J. Sci. Comput.* 50 (2012), pp. 29–62.
- [221] A. van Zuijlen and H. Bijl. “High order time integration for fluid-structure interaction on moving meshes”. In: *17th AIAA Computational Fluid Dynamics Conference*. AIAA-2005-5247. 2005.