ASC Report No. 32/2010

SDIRK Methods for the ANTARES Code

Othmar Koch, Friedrich Kupka, Bernhard Löw-Baselli,
A. Mayrhofer, F. Zaussinger

Institute for Analysis and Scientific Computing
Vienna University of Technology — TU Wien
www.asc.tuwien.ac.at ISBN 978-3-902627-03-2

Most recent ASC Reports

31/2010 *Roland Donninger, Birgit Schörkhuber, Peter C. Aichelburg*
On Stable Self-similar Blow Up for Equivalent Wave Maps: The Linearized Problem

30/2010 *Matthias Langer, Harald Woracek*
The Exponential Type of the Fundamental Solution of an Indefinite Hamiltonian System

29/2010 *Harald Woracek*
An Addendum to M.G. Krein's Inverse Spectral Theorem for Strings

28/2010 *Philipp Dörsek, Josef Teichmann*
A Semigroup Point of View On Splitting Schemes For Stochastic (Partial) Differential Equations

27/2010 *Ansgar Jüngel*
Dissipative quantum fluid models

26/2010 *Sabine Hittmeir, Ansgar Jüngel*
Cross Diffusion Preventing Blow Up in the Two-dimensional Keller-Segel Model

25/2010 *Anton Arnold, Irene M. Gamba, Maria Pia Gualdani, Stéphane Mischler, Clément Mouhot, Christof Sparber*
The Wigner-Fokker-Planck Equation: Stationary States and Large Time Behavior

24/2010 *Lehel Banjai, Christian Lubich, Jens Markus Melenk*
Runge-Kutta Convolution Quadrature for Operators Arising in Wave Propagation

23/2010 *Franz Achleitner, Sabine Hittmeir, Christian Schmeiser*
On Nonlinear Conservation Laws with a Nonlocal Diffusion Term

22/2010 *Ansgar Jüngel, Josipa-Pina Milišić*
Quantum Navier-Stokes Equations

Institute for Analysis and Scientific Computing
Vienna University of Technology
Wiedner Hauptstraße 8–10
1040 Wien, Austria

E-Mail: admin@asc.tuwien.ac.at
WWW: http://www.asc.tuwien.ac.at
FAX: +43-1-58801-10196

ISBN 978-3-902627-03-2

ASC TU WIEN

SDIRK Methods for the ANTARES Code[1]

O. Koch, F. Kupka, B. Löw–Baselli, A. Mayrhofer, and F. Zaussinger[2]

December 22, 2010

[1]The authors would like to acknowledge financial support of this work by the Austrian Science Fund FWF, project P21742-N16
[2]All: Faculty of Mathematics, University of Vienna, Nordbergstraße 15, A–1090 Wien, Austria

Contents

Chapter 1

Introduction

1.1 Motivation: Hydrodynamical flows with low viscosity and high conductivity

Many astrophysical problems can be described by mathematical models which require the numerical solution of the equations of hydrodynamics. The latter are often subject to further, analytical approximations, but may also be coupled to additional dynamical equations such as those describing radiative transfer in a medium of arbitrary optical thickness. As a particular application of this kind we mention the mathematical modelling of convective flows in stars through numerical simulations. The fluid under consideration in that case, the stellar plasma, has a very low viscosity compared to its large radiative conductivity. This property is quantified by the ratio of momentum diffusivity (kinematic viscosity) to radiative (heat) diffusivity, $\nu/\chi = \mathrm{Pr}$, where Pr is the Prandtl number of a given fluid, and here $\mathrm{Pr} \ll 1$. For stellar convection, just as for convection in the atmosphere of the Earth, viscous processes take place at length scales l_d much smaller than those which characterize large coherent structures of the flow, for instance granules or plumes. Numerical simulations of turbulent stellar convection usually aim at resolving the latter, i.e. the simulation domain with a minimum size H in each direction is chosen such that $H > L \gg h$. Here, L is a typical length scale of the mentioned spatial structures and h is the largest grid spacing anywhere in the simulation domain, $h = \max(\Delta x)$. For stellar convection simulations, we have that $h \gg l_\mathrm{d}$. Since also $\mathrm{Pr} \ll 1$, there are interesting consequences for the maximum time step Δt of such simulations.

As an example consider the numerical simulation of convection at the solar surface with a moderate resolution of about 10 to 20 grid points per vertical pressure scale height (i.e. on a radial distance $r_2 - r_1$ where $\ln(P(r_1)/P(r_2)) = 1$). The equations to be solved in this case are obtained from a spatial and temporal discretisation of the governing dynamical equations, i.e. the fully compressible Navier–Stokes equations plus associated continuity and energy equations, which are coupled to the stationary limit of the radiative transfer equation (see [32] for an example). For the moderate resolution mentioned, the maximum time step for an explicit method used for the time integration of the system of ordinary differential equations obtained from a spatial semi-discretisation has to be such that $t_\mathrm{ad} \equiv \Delta t \leq c \min((\Delta x)/a)$. Hence, the time step Δt has to satisfy the Courant–Friedrichs–Levy (CFL) condition. Thereby, c is a constant which depends on the method used for the spatial discretisation, a is either the sound speed or the maximum local flow speed in case of supersonic flow, Δx is the local grid spacing, and the minimum is taken over all grid cells of the simulation domain. Since the flow in this physical situation is close to supersonic, Δt is also close to the time scale τ at which the solution changes on the level of a few percent, i.e. the time scale on which the evolution of the solution would also be followed if an implicit time integration method were used.

However, if the resolution is increased by a factor of 5 to 10, then Δt is actually restricted by the term in the energy equation describing heat loss and gain per volume by radiative transfer, $Q_\mathrm{rad} = \mathrm{div} F_\mathrm{rad}$. In the optically thick limit, $F_\mathrm{rad} = -(\chi/(c_\mathrm{p}\rho))\nabla T$, where c_p denotes the specific heat per unit of mass, ρ is the mass density, and T the temperature. Thus, it has exactly the same form as the mathematical expression describing heat conduction in the plain heat equation. Explicit time integration methods are then subject to the analogue of the CFL condition for parabolic partial differential equations, where $t_\mathrm{rad} \equiv \Delta t \leq d(\Delta x)^2/b$, and d is some method dependent constant while b is the heat diffusivity. The point here is that below a certain threshold resolution $h_\mathrm{rad} = \Delta x$, the time step of explicit time integration methods will always be restricted by t_rad, since then $t_\mathrm{rad} < t_\mathrm{ad}$. In simulations of stellar convection, $h_\mathrm{rad} \gg l_\mathrm{d}$, a constraint which ultimately stems from $\mathrm{Pr} \ll 1$.

For simulations of stellar surface convection the most severe restriction of Δt through t_rad occurs where the dominant mechanism of energy transport in the system changes from convective to radiative (cf. the figures in [10]). This spatial region is directly

accessible to astrophysical observations and hence coincides with the transition zone where the fluid becomes optically thin (while radiation from layers underneath that region cannot be observed from outside of the object). As a result, the radiative diffusion approximation mentioned above is no longer applicable and the radiative transfer equation has to be solved. The solution of the latter converges to the diffusion approximation for layers which are optically sufficiently thick. Thus, $t_{\rm rad}$ is in general given by a less restrictive criterion ([39], see also [10] and [30] for some numerical examples). It is obtained in [39] from a linearization based analysis of the relaxation of temperature perturbations. For optically thick layers it recovers the diffusion limit and thus $\Delta t \leq d(\Delta x)^2/b$, while it predicts $t_{\rm rad}$ to be independent of the grid spacing Δx in the optically thin limit (this can readily be understood from the fact that in this case photons no longer interact with the fluid and hence cannot contribute to local heating or cooling and thus $t_{\rm rad}$ must be independent of the grid spacing). In practice, the actual time step restriction for Δt in such a simulation still decreases faster than linearly for decreasing Δx which in the end makes $t_{\rm rad}$ more restrictive than $t_{\rm ad}$ once $\Delta x \leq h_{\rm rad}$.
This restriction is even more important for stars that have surface convection zones which are subject to more efficient radiative cooling than in the case of the Sun. As a matter of fact many classes of stellar objects have this property. In that case even numerical simulations of low resolution (insufficient from both the mathematical and physical point of view) are constrained by $t_{\rm rad}$ instead of $t_{\rm ad}$. Suitable numerical methods which are more efficient under such circumstances are hence of general interest for a large parameter space in stellar astrophysical applications. [30] have analysed this restriction in details for a certain class of objects (so-called A-type stars) and found for this case that the solution of the dynamical equations evolves on the time scale $t_{\rm ad}$. This is the case even though the time step of the explicit (Runge–Kutta) time integration method used in the hydrodynamical code was clearly restricted by $\Delta t \leq t_{\rm rad} \sim 0.01\, t_{\rm ad}$ for the simulations considered in their work. From their numerical experiments they concluded that this was more akin to a problem posed by stiff differential equations than to an actual physical constraint. Thus, in principle the time step restriction might be relaxed by a suitable implicit time integration method. The physical reason for this is that for the time evolution of the energy of the fluid only the relative changes of the latter and of $Q_{\rm rad}$ are important, which are limited by the surrounding fluid once initial perturbations have relaxed on the short radiative time scales. While there is no reason to assume that this holds for numerical simulations of all classes of objects where Δt is limited by $t_{\rm rad}$ already for very low spatial resolution, the very high computational cost of astrophysical hydrodynamical simulations of surface convection for the case of efficient radiative cooling warrant a more detailed study of how to overcome the time step restrictions caused by $t_{\rm rad}$.

1.2 The ANTARES code for astrophysical simulations

We consider an implementation of simulation routines for astrophysical models realized in the ANTARES code [32]. The physical system which is simulated is the *fully compressible Navier–Stokes equation* which describes momentum conservation:

$$(\rho\mathbf{u})' + \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u} + pI) = \rho\mathbf{g} + \nabla\cdot\tau. \tag{1.1}$$

The state variables in the model equations generally depend on the spatial variables (x, y, z) and time t. The (explicit) dependencies are stated in Table 1.1. For simplicity, we omit the dependencies in the problem specification (1.1)–(1.4). The model is completed by the *continuity equation*

$$\rho' + \nabla\cdot(\rho\mathbf{u}) = 0, \tag{1.2}$$

which ensures conservation of mass, and the *total energy equation*

$$e' + \nabla\cdot(\mathbf{u}(e+p)) = \rho(\mathbf{g}\cdot\mathbf{u}) + \nabla\cdot(\mathbf{u}\cdot\tau) + Q_{\rm rad}, \tag{1.3}$$

which describes conservation of the latter. The variables and parameters which appear in the model formulation are collected in Table 1.1.
The radiative source term $Q_{\rm rad}$ is determined as the stationary limit of the *radiative transfer equation*

$$\mathbf{r}\cdot\nabla I_\nu = \rho\chi_\nu(S_\nu - I_\nu), \tag{1.4}$$

which is solved for all ray directions $\mathbf{r}$ and for all frequencies ν, resulting in the specific intensity $I_\nu(\mathbf{r})$, for details see [45]. S_ν here denotes the *source function*.
The equations of hydrodynamics (1.1), (1.2) and (1.3) are closed by the equation of state which describes the relation between the thermodynamic quantities. For the particular choice see [32].

$\rho = \rho(x, y, z, t)$	gas density
$\mathbf{u} = \mathbf{u}(x, y, z, t) = (u, v, w)^T$	flow velocity
$\rho\mathbf{u}$	momentum density
$\mathbf{u} \otimes \mathbf{u}$	Kronecker product
$p = p(T, \rho)$	gas pressure
$\mathbf{g} = (g, 0, 0)^T$	gravitational acceleration
$\tau = \tau(x, y, z, t)$	viscous stress tensor for zero bulk velocity
μ	dynamic viscosity (appears in the definition of τ)
$e = e(x, y, z, t) = e_{\text{int}} + e_{\text{kin}}$	total energy density, the sum of internal and kinetic energy densities
$T = T(x, y, z, t)$	temperature
$Q_{\text{rad}} = Q_{\text{rad}}(x, y, z, t)$	radiative source term
$\chi_\nu = \chi_\nu(T, \rho)$	(specific) opacity at frequency ν
$\kappa = \kappa(T, \rho)$	radiative conductivity
$I_\nu = I_\nu(\mathbf{r}),\ \mathbf{r} = \mathbf{r}(x, y, z)$	specific intensity along the ray of direction $\mathbf{r}$
$S_\nu = S_\nu(x, y, z)$	source function

Table 1.1: Variables and parameters in the equations (1.1)–(1.4).

For the initial condition, a slightly perturbed static model atmosphere or envelope is used which is equipped with a small seed velocity field or density perturbation to start dynamics away from equilibrium.
Boundary conditions are based on the assumption that all quantities are periodic in both horizontal directions. For the hydrodynamical equations, closed boundary conditions at the upper and lower boundary of the computational domain are used, but a recent development is to replace these by open boundary conditions. For the radiative transfer equation (1.4), incoming radiation at the boundary of the computational domain must be specified.
The ANTARES code (**A** **N**umerical **T**ool for **A**strophysical **RES**earch) described in this report solves this system of equations numerically in either one, two, or three spatial dimensions on a rectangular grid (spherical coordinates with a logarithmically rectangular grid are also possible, i.e., the grid may be locally rectangular with logarithmic grading in the radial component). ANTARES allows the definition of several grids which can be nested inside each other to improve resolution in regions of interest. At the moment, ANTARES provides up to three levels of nested grids.
For the spatial discretization of the hyperbolic terms, discretizations of ENO (*essentially non-oscillatory* [36]) type are implemented. These comprise classical ENO methods, WENO (*weighted essentially non-oscillatory*) methods [36] (optionally in conjunction with Marquina flux splitting [2]) and CNO (*convex non-oscillatory*) schemes [31]. Each of the methods uses adaptive stencils which are chosen such as to avoid spurious oscillations in the computed solution. The spatial derivatives are calculated for each direction separately.
The parabolic terms are discretized by dissipative finite difference schemes of fourth order, see Chapter 3 below. The *radiative heating rate* is determined by the *short characteristics method*, or by neans of a diffusion approximation, where appropriate, while all other source terms are evaluated at the cell centers. For the time integration, *total variation diminishing* Runge–Kutta methods are employed, see Section 1.4 below. One goal of this work is to put forward implicit SDIRK Runge–Kutta methods [8] to replace the classical explicit integrators [35, 38].
ANTARES implements two different parallelization concepts. For architectures with distributed memory, domain decomposition is used and realized by an MPI implementation. In this approach each grid is split along the horizontal direction(s) and optionally, also along vertical ones, into subdomains. The memory required to store the computational variables for each subdomain is provided by the resources available to the dedicated CPU core which performs the computations necessary for that subdomain. In this way, each CPU core is mapped to a specific geometrical volume. However, since some supercomputers offer only a limited amount of memory per CPU core and because the domain decomposition approach just mentioned is not very efficient on small grids, ANTARES offers a second type of parallelization which can be used along with or independently of the former. It is based on a shared memory concept for each subdomain and is implemented through OpenMP directives. Thus, on each MPI node or on the entire grid, if no domain decomposition is performed, the most time consuming operations which can also be performed independently of each other are identified and computed in parallel. This approach scales only to a moderate number of CPU cores (in principle up to a few dozens), but allows improvement of the scaling and the computational speed of the domain decomposition based parallelization for a larger number of problems and for a greater variety of computing architectures.

1.3 Strategies for more efficient time integration

One might be tempted to circumvent the time step restrictions associated with the requirement that a time integrator should diminish the total variation in the spatial variation through performing a smoothing procedure to lower the effective resolution and thus the constraints posed by tracing small scale variations under conditions of high conductivity, which are described in Section 1.1. However, for the problems at hand this is of limited use. While for very high resolution simulations such as those performed for the case of solar surface convection such a procedure may be viable, it is not recommendable for the case where radiative heating and cooling as represented by the function $Q_{\rm rad}$ is just barely resolved on the grid used for spatial discretisation.

This leaves us with various implicit methods at our disposal. One obvious approach in the case where $Q_{\rm rad}$ can be efficiently linearized is to use a semi-implicit method, i.e. to discretize only $Q_{\rm rad}$ with an implicit time integration method while performing the time integration of all other terms in the set of coupled partial differential equations explicitly. Such an approach is for example pursued in [21, 22]. Alternatively, the entire set of coupled equations can be integrated in time implicitly. Due to its favourable stability properties this is usually done in astrophysical applications by the θ-method. In that case the dynamical equations $\mathbf{u}'(t) = \partial\mathbf{u}(t)/\partial t = \mathbf{f}(\mathbf{u}(t))$ are approximated with the time derivative taken to be the simple difference expression $(\mathbf{u}_\mathbf{j}^{n+1} - \mathbf{u}_\mathbf{j}^{n})/(\Delta t)$ for each grid point, denoted by $\mathbf{j}$. In turn, the right-hand side is evaluated on the line between discretisations given for the old time step n and the new time step $n+1$ through $\theta\tilde{\mathbf{f}}_\mathbf{j}^{n+1} + (1-\theta)\tilde{\mathbf{f}}_\mathbf{j}^{n}$, where $\tilde{\mathbf{f}}_\mathbf{k}^{m}$ is an approximation to $\mathbf{f}$ at the point $\mathbf{k}$ for the time step m. This approximation contains the special cases of the forward Euler ($\theta = 0$) and backward Euler ($\theta = 1$) time integration schemes as well as the implicit mid-point rule ($\theta = 1/2$), but except for the latter, which is of second order, the approximation is only of first order in time. For the sake of improved robustness, θ is typically chosen larger than $1/2$, and the low accuracy which results from this approach certainly has its share in the reputation of implicit methods for hydrodynamical problems of being both costly (due to the non-linear system of equations to be solved) and inaccurate. If higher accuracy is required, a more refined approach is needed. To this end one could consider methods which are based on operator splitting and thus allow different time integration procedures to be used for the non-linear terms (the advection and pressure gradient terms in case of the hydrodynamical equations) and the molecular (or radiative) transport terms.

Alternatively, one might consider implicit time integration methods applicable simultaneously to the entire coupled set of partial differential equations. In the following we investigate this approach and study the properties of implicit time integration methods which have been designed to preserve an important stability property for higher order spatial discretisations of the non-linear advection and pressure gradient terms in the hydrodynamical equations, i.e. to force diminishing of the total variation of the solution in the course of time integration. Optimal methods have been found for this class of problems, but much less attention has been paid in previous studies to their actual efficiency when compared to explicit time integration methods near the stability limit of the latter and on their properties when applied to numerical approximations to energy fluxes caused by heat conduction, which is just the term $Q_{\rm rad}$ introduced above. We discuss these issues in detail in the following.

1.4 Total variation diminishing (TVD) time integrators

For the time integration of dissipative evolution equations whose solutions typically feature spurious oscillations in the space variables, special 'geometric integrators' which reflect the property of the original differential equation to dampen the oscillations in the course of the time propagation have been constructed.

[29] introduces the concepts of *contractivity* and *absolute monotonicity* for dissipative differential equations on general Banach spaces, gives three characterizations of the latter and shows that the two concepts are equivalent. This extends earlier work [40], where linear differential equations on Banach spaces are discussed and the concept of *radius of contractivity*, see below, is introduced. *Contractivity* of a numerical method means that for two sequences of approximations $u_n,\ \tilde{u}_n$, generated from different initial values, $\|u_n - \tilde{u}_n\| \leq \|u_{n-1} - \tilde{u}_{n-1}\|$ holds for all n in some norm. Similarly, *monotonicity* means that $\|u_n\| \leq \|u_{n-1}\|$ [5]. Monotonic numerical schemes are also referred to as *strong stability preserving*. Clearly, the two concepts correspond for linear problems. A general characterization of monotonicity is given in [5], together with specializations to some particular norms. Note that definitions referring to the interval ends or internal stages have been proven to be equivalent [13]. If the used norm is the *total variation norm* $\|\cdot\|_{TV}$, the property is commonly referred to as *total variation diminishing*. The weaker requirement of *total variation boundedness*, $\|u_n\|_{TV} \leq M,\ M > 1$, holds for some spatial discretisations where TVD is violated [7]. This yields an analogous theory. Under some additional assumption, boundedness implies monotonicity, however, see [26]. If the assumption is violated, a counterexample for a general norm is also given in that paper.

In [29], *unconditional contractivity*, which holds for all step sizes, is shown to limit the stage order $\tilde{p}$ and the global order p of general Runge–Kutta methods to 1 (recall $\tilde{p} \leq p$), see also [40]. For finite *radius of contractivity* $R(A,b) > 0$ (later called *Kraaijevanger's coefficient* which corresponds to the optimal *CFL number* c of a Runge–Kutta method, see below), the stage order of a Runge–Kutta scheme with Butcher array $(A,b)^T$ is limited to 2 and the global order to 6. This bound is sharp, i.e., a fully implicit sixth order method has been constructed in [28]. For explicit Runge–Kutta methods, the stage order is at most 1 and global order at most 4 in general (for linear problems, however, explicit TVD methods of any order can be constructed [13]). Generally, the Kraaijevanger coefficient is limited by $R < s - p + 1$ for an explicit Runge–Kutta method of order p with s stages. Furthermore, many contractive ERK schemes are derived in [29] such that the theoretical optimum on $R(A,b)$ is attained:

- For $p = 1$, $R \leq s$ holds. A method with $R = s$ is derived, which is unique.
- For $p = 2$, $R \leq s - 1$ holds. Methods with $R = s - 1$ are derived, which are unique. For $s = 2$, this corresponds with the Osher/Shu method of second order put forward earlier [35].
- For $p = 3$, $R \leq 1$ holds for $s = 3$. A method with $R = 1$ is derived, which is unique. This corresponds with the Osher/Shu method of third order [35]. For $s = 4$, $R \leq 2$ is established. A method with $R = 2$ is derived, which is unique in this class.
- For $p = 4$, no scheme with $s = 4$ exists such that $R > 0$. For $s = 5$, coefficients are determined numerically, which suggest that $R \approx 1.5$.

[29] shows that a necessary condition for $R > 0$ is that all entries of the coefficient matrix A are nonnegative and integration weights b_j positive. Otherwise, the right-hand side operators must be modified, which generally leads to higher computational cost [38], which can be avoided in some situations, however [14]. This modification can be interpreted as a perturbation [20]. Moreover, it was demonstrated in [20] that the case where an Osher/Shu representation with all positive coefficients exists corresponds with $R(A,b) > 0$. Consequently, the positivity of the coefficients is guaranteed for all the methods we consider, if a suitable representation is chosen. Conversely, if A is nonnegative, the stage order is limited by 2, where equality holds only if A has a zero row [28]. An important practical aspect of implicit methods is the existence of a unique solution to the non-linear algebraic equations. In [29] it is demonstrated that this problem is well-posed under the same time-step restriction as for the TVD property.

The first discussion of (explicit) TVD Runge–Kutta methods is given in [35, 38]. This class of methods is further developed in [15]. [35] derives a particular *Osher/Shu representation* for studying the TVD property for explicit Runge–Kutta methods under a CFL condition defined in terms of the resulting coefficients. It should be noted however that this Osher/Shu representation is not unique, see below. [35] develops the corresponding concepts also for (explicit) linear multistep methods. In [38], explicit methods with CFL numbers equal to 1 are constructed for orders up to $p = 3$, which can be shown to be optimal for the respective number of stages in the class of explicit Runge–Kutta methods of these orders [15]. In these early works on explicit TVD methods, a fixed Osher/Shu representation is put forward, which possibly yields negative coefficients even for methods with positive CFL numbers. It was later shown however that for all Runge–Kutta methods with positive CFL numbers, an Osher/Shu representation exists with all positive coefficients [4]. [35] also derives explicit TVD Runge–Kutta methods of orders $p = 2, 3, 4$ for non-linear problems and a class of fifth-order schemes for linear problems. [15] later conducted a more systematic study and found optimal schemes with $p = 2,\ s = 2$ and $p = 3,\ s = 3$, respectively, and additionally a low-storage scheme with $p = s = 3$. A review and further development of the theory is given in [16], where optimal explicit TVD Runge–Kutta methods for linear problems are derived and a systematic study of explicit multi-step methods is conducted. [42] introduces a new class of methods of orders up to $p = 4$ with $s > p$, where for $p = 2,\ s = 3$ a method with CFL number $c = 2$ results, for $p = 2,\ s = 4$, $c = 3$ is realized, and for $p = 3,\ s = 4$, a method with $c = 2$ is found. [34] gives order barriers for TVD methods based on a fixed Osher/Shu representation. In [12] optimal Runge–Kutta methods with $s > p$ are derived, while for $p = s = 2$ and $p = s = 3$ optimal methods with $c = 1$ had earlier been found in [15, 38]. [42] gives a method with $p = 4,\ s = 5$ and $c \approx 1.5$. For large scale computations, low storage schemes are of particular interest, these are discussed in [15, 47].

[4] extends the *Osher/Shu representation* [36] and associated analysis to general implicit Runge–Kutta methods. Note that the Osher/Shu representation of a general Runge–Kutta method is not unique. Thus, for a particular representation a step-size limitation c for monotonicity is introduced, which is commonly referred to as *CFL number*. It turns out that $R(A,b)$ is the maximum of c over all Osher/Shu representations [4]. $R(A,b)$ is thus a property of the numerical method independent of the particular representation of the Runge–Kutta method. [6] gives three procedures for constructing methods with optimal step-size coefficients.

Very successful Runge–Kutta methods which are cheap to implement are *singly diagonally implicit Runge–Kutta methods* (SDIRK) [1]. [4] first introduces a TVD SDIRK method with $p = s = 2$, which was later proven to be optimal [8]. The latter paper gives a comprehensive theory of TVD SDIRK methods. It is shown that for SDIRK methods, $R = c$ holds, which more generally holds for irreducible methods (where the rows of A are distinct). There are no TVD SDIRK methods with $R > 0$ for $p > 4$. Higher order has the disadvantage that for the SDIRK methods, R decreases with the order p of the methods. The optimal SDIRK methods are listed in [8] as follows:

- For $p = 1$, the optimal method with $R = \infty$, consists of s repeated applications of the backward Euler method.
- For $p = 2$, the methods with $s = 1, 2, 3$ have been proven to be optimal, see [4, 28]. For $s > 3$, the methods [8] are conjectured to be optimal in the class of SDIRK methods based on numerical optimization. An extensive numerical search in [28] confirms that the methods are even optimal in the class of all implicit Runge–Kutta methods.
- For $p = 3$, the SDIRK methods [8] were numerically found to be optimal in the class of SDIRK methods, and conjectured to be optimal even in the class of all implicit Runge–Kutta methods [28].
- For $p = 4$, three SDIRK methods are known for $s = 3$, one of which is TVD. For $s = 4, \ldots, 8$, optimal methods are determined numerically in [8], while for $p > 4$ no SDIRK methods exist with $R > 0$.

It should be noted that the CFL numbers for SDIRK methods decrease with increasing order $p = 2, 3, 4$.
Note that [13] gives optimal methods from other classes of Runge–Kutta methods as well: For $p = 4, \ldots, 6$, the optimal schemes are *Diagonally Implicit (DIRK)* schemes; for DIRK methods with s stages, $p \leq s + 1$, and the same holds for SIRK methods with $R > 0$ [28]. For *Singly Implicit (SIRK)* methods with $p \geq 5$, the step-size coefficient has the same size as for the optimal explicit method of the same order.
[20, 21, 22] specialize the analysis of strong stability preserving methods to *additive* or *implicit/explicit (IMEX) Runge–Kutta methods*. A characterization of monotone methods in this class is given in terms of the coefficient matrices and the order barrier $p \leq 4$ is established. However, no comprehensive search for optimal methods is conducted.
[25] on the other hand studies total variation diminishing and total variation boundedness for linear multistep methods, which was first discussed in [35], where a canonical representation is put forward which allows conclusions on the CFL number from the method's coefficients. [16] shows that for a multistep method with nonnegative coefficients, the number of steps has to be larger than the order. A number of optimal schemes are derived in [16, 35]. For this class of methods it is furthermore found in [13] that implicit methods provide no advantage over explicit methods. Moreover, for explicit methods, the step-size coefficient satisfies $c < 1$, and for implicit methods $c \leq 2$ [13]. Finally, an increase in the number of steps in a multistep method does not increase the computational cost, but induces higher storage demand [37].
[41] extends the theory of monotone discretisations to general numerical processes, and in particular to the class of *general linear methods* [18], which comprises Runge–Kutta and linear multistep methods. A nontrivial extension of this work is given in [23]. Monotonicity for general linear methods in seminorms or even sublinear functionals is studied in [24]. It is demonstrated in [13] that for explicit methods in this class, the step-size coefficient c is bounded by the number of internal stages.
Practical aspects of TVD integrators are discussed in [13]. The computational advantages of using TVD methods are demonstrated by means of an example, and low storage implementations [27] are reviewed. Moreover, the efficiency of the numerical methods is assessed, where the ratio of step-size coefficient and number of stages per interval of the corresponding length serves as the indicator. It is found that SDIRK with $p = 2$ is optimal in that respect, while this quantity decreases with the order of the method. However, a larger number of stages generally leads to more efficient methods. The values are considerably higher than for explicit methods [13], but of course this will be compensated for to some extent by the cost of the solution of the non-linear algebraic equations. In fact, the optimal implicit methods excel by factors of 3–4, which sets the target for the number of required fixed point iterations.

Chapter 2

SDIRK methods

We want to solve numerically the (autonomous) initial value problem

$$y'(t) = F(y(t)), \quad y(0) = y_0. \tag{2.1}$$

Consider a general implicit s-stage Runge–Kutta method

$$y_i = y_{\text{old}} + \Delta t \sum_{j=1}^{s} a_{i,j} F(y_j), \quad i = 1, \ldots, s, \tag{2.2}$$

$$y_{\text{new}} = y_{\text{old}} + \Delta t \sum_{j=1}^{s} b_j F(y_j). \tag{2.3}$$

The condition for the convergence of the fixed point iteration for the solution of the associated non-linear algebraic equations from [11, p. 40] is

$$\Delta t < \frac{1}{L \max_{i=1,\ldots,s} \sum_{j=1}^{s} |a_{i,j}|}, \tag{2.4}$$

where L denotes a Lipschitz constant for F. Ferracina & Spijker [3, 8] introduce the following classes of *Singly Diagonally Implicit Runge–Kutta (SDIRK)* methods with the TVD property of convergence orders $p = 2$ and $p = 3$, respectively:
$\underline{p = 2}$:

$$\begin{aligned} a_{i,j} &= \begin{cases} \frac{1}{2s}, & i = j,\ 1 \le i \le s, \\ \frac{1}{s}, & 1 \le j < i \le s, \\ 0, & \text{otherwise}, \end{cases} \\ b_j &= \frac{1}{s}, \quad j = 1, \ldots, s. \end{aligned}$$

The CFL numbers for these schemes were proven to be equal to $2s$ for each $s \ge 1$.
Condition (2.4) for the radius of convergence of the fixed point iteration translates to

$$\Delta t < \frac{1}{L} \frac{2s}{2s-1}.$$

Upon closer inspection of the analysis in [11, p. 40], this estimate can be improved for the SDIRK methods: Their special structure allows to compute the approximations for the stages consecutively, i.e., when discussing the iteration error for the i-th stage we can assume that values for the former stages are given and not subject to further change. Hence, for the i-th stage let

s	2	3	4	5	6	7	8	9	10
CFL	2.732	4.828	6.873	8.899	10.916	12.928	14.937	16.944	18.950

Table 2.1: CFL numbers for the third order SDIRK

$\varepsilon_i^{(m)} = y_i - y_i^{(m)}$ denote the error of the m-th iterate from the fixed point, i.e. the exact solution of the implicit scheme. Then

$$\begin{aligned}\left|\varepsilon_i^{(m+1)}\right| &= \Delta t \left| y_{\text{old}} + \Delta t \sum_{j=1}^{s} a_{i,j} F(y_j) - \left(y_{\text{old}} + \Delta t \sum_{j=1}^{s} a_{i,j} F(y_j^{(m)}) \right) \right| \\ &\leq \Delta t L \sum_{j=1}^{s} |a_{i,j}| \left|\varepsilon_j^{(m)}\right| \\ &= \Delta t L ||a_{i,i}| \left|\varepsilon_i^{(m)}\right|.\end{aligned}$$

The iteration is thus contractive if

$$\Delta t < \frac{1}{L \max_{i=1,\ldots,s} |a_{i,i}|} = \frac{2s}{L}. \tag{2.5}$$

$\underline{p=3}$:

$$\begin{aligned} a_{i,j} &= \begin{cases} \frac{1}{2}\left(1 - \sqrt{\frac{s-1}{s+1}}\right), & i = j,\ 1 \leq i \leq s, \\ \frac{1}{\sqrt{s^2-1}}, & 1 \leq j < i \leq s, \\ 0, & \text{otherwise}, \end{cases} \\ b_j &= \frac{1}{s}, \quad j = 1, \ldots, s. \end{aligned}$$

The CFL numbers were shown to be equal to $s - 1 + \sqrt{s^2 - 1}$, which corresponds to the numerical values specified in Table 2.1. For these schemes, the condition (2.4) for the radius of convergence of the fixed point iteration translates to

$$\Delta t < \frac{2}{L} \frac{1}{1 + \sqrt{\frac{s-1}{s+1}}}.$$

If we refine the estimate analogously to (2.5), we obtain

$$\Delta t < \frac{1}{L \max_{i=1,\ldots,s} |a_{i,i}|} = \frac{2}{L} \frac{1}{1 - \sqrt{\frac{s-1}{s+1}}}. \tag{2.6}$$

However, it may be inadvisable to choose the step-size too close to the boundary of the circle of convergence just derived. From the Banach fixed point theorem, clearly [17]

$$\left|\varepsilon_i^{(m)}\right| \leq \frac{q^m}{1-q} \left|\varepsilon_i^{(1)}\right|, \tag{2.7}$$

where

$$q := \Delta t L \max_{i=1,\ldots,s} \sum_{j=1,\ldots,s} |a_{i,j}|. \tag{2.8}$$

For the SDIRK methods, the simplifications as in (2.5), (2.6) apply correspondingly, so that

$$q = \Delta t \frac{L}{2s} \qquad \text{for } p = 2, \tag{2.9}$$

$$q = \Delta t \frac{L}{2} \left(1 - \sqrt{\frac{s-1}{s+1}}\right) \qquad \text{for } p = 3. \tag{2.10}$$

Thus, the convergence of the fixed point iteration becomes intolerably slow as $q \to 1$, and we should choose the step-size Δt such as to ensure convergence in only a few steps.
To enlarge the step-sizes which still enable successful iterative solution of the numerical solution at the stages and provide a good starting guess for the iteration, we propose to use a predictor step given by the forward Euler method in each stage. It turns out that this predictor can be realized without additional evaluation of the right-hand side or additional memory requirement, see Section 2.3 below.

2.1 Stability regions

We have investigated the stability regions of the SDIRK schemes. The *stability function* of a Runge–Kutta scheme is defined by

$$R(z) = 1 + zb^T(I - zA)^{-1}\mathbf{1}, \tag{2.11}$$

where $b = (b_1, \ldots, b_s)^T$, $A = (a_{i,j})_{i,j=1}^s$, $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^s$, and I is the identity matrix in $\mathbb{R}^{s\times s}$. The method is called *A-stable* if $|R(z)| < 1 \forall z \in \mathbb{C}^- = \{z \in \mathbb{C} : \Re(z) < 0\}$. It is called *L-stable* if additionally $\lim_{|z|\to\infty} R(z) = 0$.
A MAPLE calculation (performed for several values of s) shows that for the scheme with $p = 2$,

$$R(z) = (-1)^s \left(\frac{2s + z}{-2s + z}\right)^s,$$

whence the methods are A-stable, but not L-stable. The stability region coincides with $\mathbb{C}^-$. Moreover, we checked the proposition [19, Theorem 2.2] that

$$\exp(z) - R(z) = O(z^{p+1}) \tag{2.12}$$

for a Runge-Kutta method of order p. Indeed, our MAPLE calculation shows that

$$\begin{aligned}
\exp(z) - R(z) &= -\frac{1}{48}z^3 + O(z^4), \quad s = 2,\\
\exp(z) - R(z) &= -\frac{1}{108}z^3 + O(z^4), \quad s = 3,\\
\exp(z) - R(z) &= -\frac{1}{192}z^3 + O(z^4), \quad s = 4,\\
\exp(z) - R(z) &= -\frac{1}{300}z^3 + O(z^4), \quad s = 5,\\
\exp(z) - R(z) &= -\frac{1}{432}z^3 + O(z^4), \quad s = 6.
\end{aligned}$$

Likewise, the stability functions for the schemes with $p = 3$ were computed by MAPLE:

$$\begin{aligned}
R(z) &= \frac{3(z + 1 + \sqrt{3})^2}{(3 - z + \sqrt{3})^2}, \qquad s = 2,\\
R(z) &= \frac{2\sqrt{2}(z + 2 + 2\sqrt{2})^3}{(4 - z + 2\sqrt{2})^3}, \qquad s = 3,\\
R(z) &= \frac{25(z + 3 + \sqrt{15})^4}{9(5 - z + \sqrt{15})^4}, \qquad s = 4,\\
R(z) &= \frac{9\sqrt{6}(z + 4 + 2\sqrt{6})^5}{8(6 - z + 2\sqrt{6})^5}, \qquad s = 5,\\
R(z) &= \frac{343(z + 5 + \sqrt{35})^6}{125(7 - z + \sqrt{35})^6}, \qquad s = 6.
\end{aligned}$$

s	$z_{\min}$
2	$-6-4\sqrt{3}\approx -12.93$
3	≈ -37.10
4	$-30-8\sqrt{15}\approx -60.98$
5	≈ -101.03
6	≈ -140.99

Table 2.2: Left boundary of stability regions for $p = 3$.

This seems to correspond to

$$R(z) = C\left(\frac{z+s-1+\sqrt{s^2-1}}{s+1-z+\sqrt{s^2-1}}\right)^s,$$

with $C = \left(\frac{s+1+\sqrt{s^2-1}}{s-1+\sqrt{s^2-1}}\right)^s$. We also checked (2.12) with MAPLE and found that this holds for $s = 2, \ldots, 6$ (the constants for the leading terms are too awkward to specify).
The stability regions for these five values of s are plotted in Figure 2.1. The methods are *not A-stable*, and unfortunately have only a bounded stability region. However, the stability region appears to become bigger as s increases. These methods are not L-stable either.

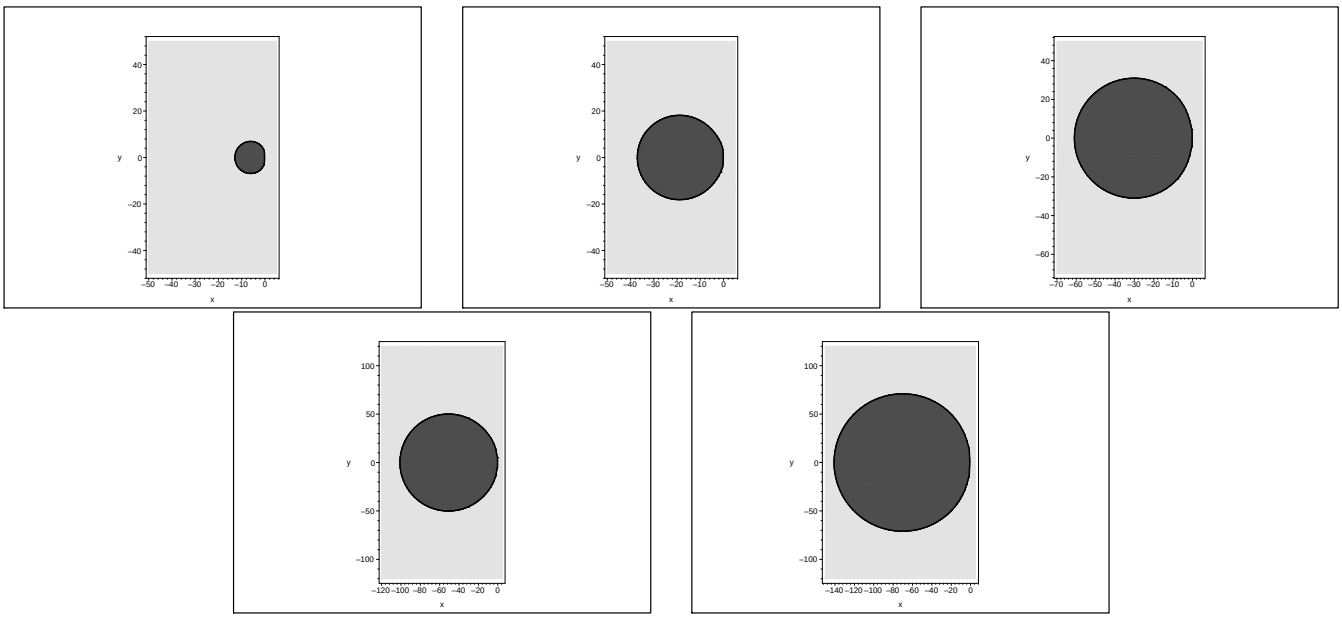


Figure 2.1: Stability regions of third order methods with $s = 2, \ldots, 6$.

We evaluate numerically the point $z_{\min}$ where the stability region intersects the real axis in the left half-plane, see Table 2.2 for the results, where the cases $s = 3, 5, 6$ could only be approximated numerically.

Validation of the implementation

To check the correctness of our realization of the SDIRK methods which is described in detail in Section 2.3 below, we compute the empirical convergence orders of the methods. We solve the non-linear test problem

$$y'(t) = 1 + y^2(t), \qquad y(0) = 0, \tag{2.13}$$

Δt	error	order	C
1.6250e−01	2.0922e−01	—	—
8.1250e−02	4.5014e−02	2.22	1.1744e+01
4.0625e−02	1.0890e−02	2.05	7.6788e+00
2.0313e−02	2.7010e−03	2.01	6.8460e+00
1.0156e−02	6.7391e−04	2.00	6.6194e+00
5.0781e−03	1.6840e−04	2.00	6.5548e+00
2.5391e−03	4.2093e−05	2.00	6.5365e+00
1.2695e−03	1.0523e−05	2.00	6.5316e+00

Table 2.3: Empirical convergence order for SDIRK 2 with $s = 1$ applied to (2.13).

Δt	error	order	C
6.5000e−01	8.2939e−02	—	—
3.2500e−01	1.9523e−02	2.09	2.0379e−01
1.6250e−01	4.8116e−03	2.02	1.8916e−01
8.1250e−02	1.1987e−03	2.01	1.8391e−01
4.0625e−02	2.9941e−04	2.00	1.8215e−01
2.0313e−02	7.4835e−05	2.00	1.8160e−01
1.0156e−02	1.8708e−05	2.00	1.8144e−01
5.0781e−03	4.6767e−06	2.00	1.8141e−01
2.5391e−03	1.1690e−06	2.00	1.8157e−01
1.2695e−03	2.9212e−07	2.00	1.8205e−01

Table 2.4: Empirical convergence order for SDIRK 2 with $s = 6$ applied to (2.13).

with the known exact solution $y(t) = \tan(t)$. Tables 2.3 and 2.4 give the stepsizes Δt, exact errors and empirical convergence order at $t = 1.3$ computed as

$$\text{order} = -\log\left(\frac{|y_N^h - y(1.3)|}{|y_{2N}^{h/2} - y(1.3)|}\right) / \log(2),$$

for the SDIRK 2 methods with $s = 1$ and $s = 6$, respectively, when applied to (2.13). The last column gives the error constant C. Note that as required this converges to a constant independent of Δt.
Table 2.6 gives the same information for SDIRK 3 with $s = 6$.
As later on we want to compare the accuracy of the SDIRK methods with that of the Osher/Shu methods, we enable a fair comparison between the schemes by repeating the experiment above for the cases $p = 2,\ s = 1$ and $p = 3,\ s = 2$. The results are given in Tables 2.3 and 2.5. Note that the results for the coarsest step-sizes are missing because in these cases the fixed point iteration did not converge.
Referring to the error constant C, which has a great influence on the accuracy of the method, we investigate the dependence on the number s of stages in the SDIRK methods. Indeed, the error constant is smaller for larger s. In Table 2.7, we give the error constants we observed in the step for $h = 5.0781 \cdot 10^{-3}$ for $p = 2,\ 3$ and $s = 2, \ldots, 6$. We observe that the error constant decreases with s and is smaller for the methods with $p = 3$. The decrease in the error constants with growing s can easily be seen to be quadratic (this relationship holds up to 3% deviation in all cases except $p = 3,\ s = 2$).

2.2 Implications of the convergence radius of the fixed point iteration

In the following we investigate in more detail time step restrictions on SDIRK methods, if we solve the non-linear system (2.2)–(2.3) by fixed point iteration. We also consider the case in which the right-hand side $F(y(t))$ results from spatial discretisations of the one-dimensional heat equation and of the one-dimensional advection equation, respectively.

Δt	error	order	C
6.5000e−01	4.3959e−01	—	—
3.2500e−01	4.6173e−02	3.25	1.7835e+00
1.6250e−01	5.9921e−03	2.95	1.2657e+00
8.1250e−02	7.5890e−04	2.98	1.3493e+00
4.0625e−02	9.4520e−05	3.01	1.4335e+00
2.0313e−02	1.1742e−05	3.01	1.4505e+00
1.0156e−02	1.4611e−06	3.01	1.4375e+00
5.0781e−03	1.8199e−07	3.01	1.4282e+00
2.5391e−03	2.2583e−08	3.01	1.4694e+00
1.2695e−03	2.6370e−09	3.10	2.4814e+00

Table 2.5: Empirical convergence order for SDIRK 3 with $s = 2$ applied to (2.13).

Δt	error	order	C
6.5000e−01	3.0890e−02	—	—
3.2500e−01	4.3508e−03	2.83	1.0444e−01
1.6250e−01	5.6618e−04	2.94	1.1873e−01
8.1250e−02	7.0568e−05	3.00	1.3295e−01
4.0625e−02	8.7146e−06	3.02	1.3747e−01
2.0313e−02	1.0788e−06	3.01	1.3592e−01
1.0156e−02	1.3394e−07	3.01	1.3372e−01
5.0781e−03	1.6541e−08	3.02	1.3853e−01

Table 2.6: Empirical convergence order for SDIRK 3 with $s = 6$ applied to (2.13).

$p\backslash s$	$s = 1$	$s = 2$	$s = 3$	$s = 4$	$s = 5$	$s = 6$
$p = 2$	6.55e+00	1.64e+00	7.26e−01	4.09e−01	2.62e−01	1.82e−01
$p = 3$	—	1.43e+00	5.65e−01	3.09e−01	1.97e−01	1.39e−01

Table 2.7: Error constants for SDIRK 2 and SDIRK 3 with $s = 2, \ldots, 6$ applied to (2.13).

Consider first the forward or explicit Euler method for the time integration of the simple linear equation $\dot{y} = -\lambda y$, with $\lambda > 0$ a real constant. In this case, $y^{n+1} = y^n + \Delta t(-\lambda y^n) = (1 - (\Delta t)\lambda)y^n$, thus $|y^{n+1}| \leq |1 - \lambda\Delta t||y^n|$ and hence $|1 - \lambda\Delta t| \leq 1$ for stability, which is equivalent to $0 \leq \lambda\Delta t \leq 2$. If in addition we require y^{n+1} not to change sign, we also have to demand that $0 < 1 - \lambda\Delta t$, whence monotonicity of y^{n+1} is obtained under the restriction $\lambda\Delta t < 1$. Thus, for the forward Euler method we have to impose

$$\Delta t \leq \frac{2}{\lambda} \qquad \text{for stability and} \tag{2.14}$$

$$\Delta t < \frac{1}{\lambda} \qquad \text{for monotonicity.} \tag{2.15}$$

An implicit Runge–Kutta method applied to $\dot{y} = -\lambda y$ with $\lambda > 0$ in general will have a much larger stability domain than (2.14), as we have shown in Section 2.1 for the SDIRK methods defined further above, but the convergence radius of the fixed point iteration used to solve the resulting non-linear system of equations introduces similar restrictions as in (2.14)–(2.15). For the SDIRK methods of orders $p = 2$ and $p = 3$ the respective convergence radii are given by (2.5) and (2.6). Depending on the specific SDIRK method, the constraints (2.5) and (2.6) may not only imply stability, but also monotonicity (see Section 3.3).
We now discuss some relations between (2.14)–(2.15) and the convergence radius (2.5), respectively (2.6), of the fixed point iteration when applied to SDIRK methods. We first note that each stage of an SDIRK method, if solved by fixed point iteration, requires consecutive forward Euler steps of size $\Delta t/(2s)$ for the s-stage SDIRK method with $p = 2$, where Δt is the time step chosen for the entire Runge–Kutta step. Now from (2.5) we have the restriction $\Delta t < 2s/L$. Hence, each iteration step satisfies $\Delta t < 1/L$ which is just the requirement of monotonicity (2.15) for the forward Euler method with $L = \lambda$. The same is valid for the s-stage SDIRK method with $p = 3$, where the iteration step in each stage is a forward Euler step with $\Delta t\,(1 - \sqrt{(s-1)/(s+1)})/2$. Again, this satisfies the monotonicity requirement (2.15), if the time step Δt is restricted as in (2.6) with $L = \lambda$. The convergence radius (2.5), respectively (2.6), hence ensures that the forward Euler steps of the fixed point iteration are monotonic and stable.
Another important property is found for the first iteration within consecutive stages for methods where $s > 1$. In such a case, for the SDIRK method with $p = 2$, the tableau of coefficients $a_{i,j}$ implies that we can use a predictor step of length $\Delta t/s$ to advance from the solution y_{i-1} at the previous stage $i - 1$ to a first estimate of the solution y_i at the current stage i with $i > 1$. Clearly, $\Delta t/s \leq 2/L$ for such a predictor step with Δt restricted by (2.5). Hence, the predictor step is within the stability range (2.14) of the forward Euler method. This is not the case for the SDIRK method with $p = 3$ and the time step Δt taken near the limit given by (2.6), since its tableau of coefficients $a_{i,j}$ requires a predictor step of length $\Delta t/\sqrt{s^2 - 1}$. Thus the stability constraint (2.14) with $L = \lambda$ cannot be fulfilled with $\Delta t = (2/L)/(1 - \sqrt{(s-1)/(s+1)})$, because $s > \sqrt{s^2 - 1}$ for any positive integer s. Thus, the predictor step would actually have to be chosen slightly smaller, i.e. just as large as for the s-stage SDIRK method with $p = 2$. However, this is of limited practical relevance, since close to the convergence radius ($q \to 1$) given by either (2.9) or (2.10) for the SDIRK methods with $p = 2$ and $p = 3$, respectively, the convergence of the fixed point iteration is intolerably slow, see (2.7). It can easily be verified that for an acceptable convergence rate with, say, $q < 0.7$, also the predictor step for the SDIRK method with $p = 3$ is always within the stability range given by (2.14).
In Section 3.2 below we discuss the dissipativity of various schemes for solving the heat equation including the standard forward in time, centred in space scheme (3.3)–(3.4), where we recall that for $\mu := b\Delta t/(\Delta x)^2$ the scheme is dissipative if $\mu < 1/2$, and monotonic if $\mu < 1/4$. For stability of the approximation it is sufficient that $\mu \leq 1/2$. Proofs of these results can be found in the discussion of this scheme in [43]. We may hence define $C_\mathrm{d} := \mu$ as the (diffusive) Courant number. If we apply the s-stage SDIRK method with $p = 2$ to solve the heat equation for the spatial discretisation (3.4), there is no stability constraint from the SDIRK method itself due to its A-stability. Hence, the time step limitation when using fixed point iteration for solving the resulting system of equations is exclusively due to the iteration itself. For fixed b, Δt, and Δx it is easy to see that the time step limitation is equivalent to that obtained when integrating $\dot{y} = (-4b/(\Delta x)^2)\, y$ with the forward Euler scheme and $\lambda = 4b/(\Delta x)^2 = L$. From the previous discussion we recall that the s-stage SDIRK method with $p = 2$ has a step width $2s$ times the size of an explicit Euler time step, which is monotonic provided that $\max C_\mathrm{d} < 1/4$, and we conclude that

$$\max C_\mathrm{d} = \frac{s}{2}. \tag{2.16}$$

By the same reasoning for the s-stage SDIRK method with $p = 3$ we find that

$$\max C_\mathrm{d} = \frac{1}{2 - 2\sqrt{\frac{s-1}{s+1}}}. \tag{2.17}$$

For $s = 2$ this yields $\max C_\mathrm{d} \approx 1.183$ for the SDIRK method with $p = 3$, which is slightly larger than the value of $\max C_\mathrm{d} = 1$ obtained for $p = 2$.
It is possible to repeat this reasoning for the linear, constant coefficient advection equation with advection velocity $a > 0$, if we consider spatial discretisations suitable for a time integration which is TVD with positive step size coefficient. This includes the standard first order upwind differencing, for which $0 \le a\Delta t/\Delta x \le 1$. Thus, the maximum (advective) Courant number $C_\mathrm{a} = a\Delta t/\Delta x$ is 1 for this scheme. We note that the time integration in this case can be compared to a linear equation $\dot{y} = \lambda y$ with complex λ, where the real part of λ is negative due to the numerical dissipation of the scheme introduced by requiring the numerical approximation to have the TVD property. Since $\max C_\mathrm{a} = 1$ relates to the stability boundary for the forward Euler scheme, we conjecture that

$$\max C_\mathrm{a} = s \tag{2.18}$$

for the SDIRK scheme with $p = 2$ and s stages and

$$\max C_\mathrm{a} = \frac{1}{1 - \sqrt{\frac{s-1}{s+1}}} \tag{2.19}$$

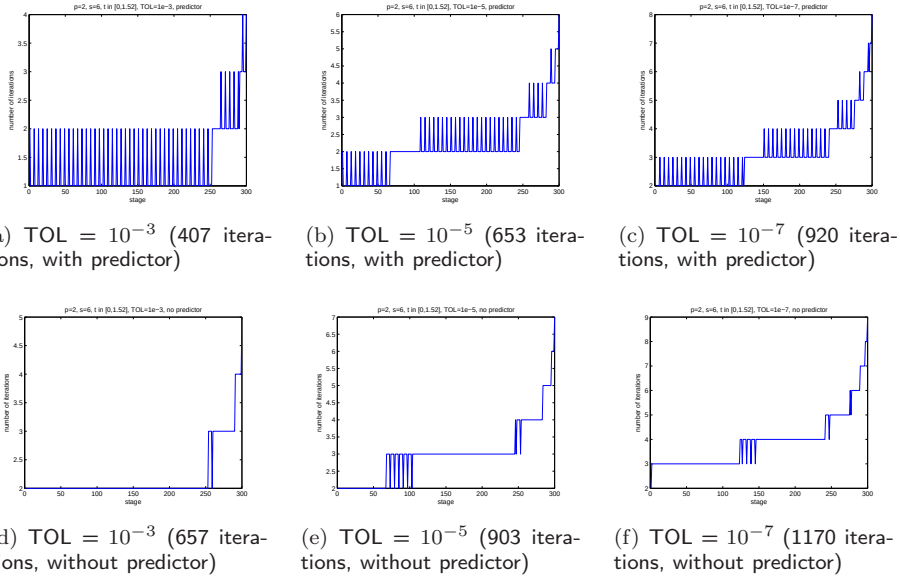
for the s-stage SDIRK scheme with $p = 3$. For each of these methods $\max C_\mathrm{a}$ is less than their CFL number as listed further above, but in no case worse than that by a factor of $1/2$. The restriction is less rigorous than the diffusive limitation in the scenario considered just above.
The reasoning above can also be used to assess the expected performance of the SDIRK methods: the forward Euler method is monotonic if $C_d < 1/4$, while for the second-order Osher/Shu method $C_d < 1/2$ should hold and SDIRK 2 has this property for $C_d < s/2$. The forward Euler method requires 1 evaluation of the right-hand side per interval, the second-order Osher/Shu method 2 evaluations and the second-order SDIRK method $s\cdot$(# fixed point iterations). Thus, if two fixed point iterations are required to compute each stage, the efficiency of the methods should be comparable. For the third order methods, by the same reasoning, the gain is $\frac{1}{2s}\frac{1}{1-1\sqrt{\frac{s-1}{s+1}}}$. For $s = 2$, the gain in efficiency can thus be estimated to be about 18%.
Finally, to illustrate the considerations on the radius of convergence of the fixed point iteration from Section 2.2, we have tested the maximal radius of convergence for the test problem (2.13). Table 2.8 gives for the SDIRK methods with $p = 2,\ s = 1, \ldots, 6$, and $p = 3,\ s = 2, \ldots, 6$, the largest step-size where the iteration successfully converged, i.e., an empirical value for the radius of convergence. The first column gives the point t^* corresponding to the rightmost stage where the iteration was convergent. We observed that a possible failure to converge always occurred at the last stage in the interval $[0, 1.52]$. The second column gives the corresponding solution value $\tan(t^*)$, the third column the Lipschitz constant L which in our case is computed as $L = 2\tan(t^*)$. The fourth column gives the error at t^*, and the fifth and sixth columns contain the number N of intervals in the coarsest mesh with a successful iteration and corresponding step for the fixed point iteration. Finally, we compare our experimental results for the convergence radius with the theoretically predicted values (2.5) and (2.6), which are given in the last column, scaled to the steplength of the intermediate stage such as to obtain the correct quantity to compare to the computed value. The predicted value ρ for $p = 2$ is equal to $\frac{1}{s}\frac{2s}{L} = \frac{2}{L}$, according to (2.5) and recalling that the distance between two stages is $\frac{\Delta t}{s}$. With the same reasoning, for $p = 3$ we compare the predicted value $\rho = \frac{2}{L\sqrt{s^2-1}}\frac{1}{1-\sqrt{(s-1)/(s+1)}}$ with the actual step on the coarsest grid, $\frac{\Delta t}{\sqrt{s^2-1}}$.
We observe that for the method of second order, the convergence radius is almost independent of the number of stages s. The prediction overestimates the convergence radius by about 70%.
For the third order methods, the radius is larger than for the second-order methods by about 20–30%, largest overall for $p = 3,\ s = 2$ as predicted in Section 2.2, and decreases with larger s.

Number of fixed point iterations

We now study the number of fixed point iterations required for the solution of the Runge–Kutta equations for (2.13) on the interval $t \in [0, 1.52]$ with $N = 50$ steps, comparing the variant with the predictor or without the predictor, where the approximation computed for one stage is used as the starting value of the fixed point iteration for the next stage. The results are given first for the case $p = 2,\ s = 6$ for iteration tolerances $10^{-3},\ 10^{-5},\ 10^{-7}$ in Figure 2.2. The total numbers of iterations are given in brackets, respectively. We observe that the number of iterations is reduced by the use of the predictor.
We repeat the same experiment with the same problem data and parameters for the numerical method with $p = 3,\ s = 6$. The results are given in Figure 2.3. Again, the predictor is advantageous.

	t^*	$\tan(t^*)$	L	Error	N	step	ρ
SDIRK $p=2,\ s=1$	1.4976e+00	1.3646e+01	2.7293e+01	8.2464e+00	34	4.4706e−02	7.3279e−02
SDIRK $p=2,\ s=2$	1.4976e+00	1.3646e+01	2.7293e+01	8.2465e+00	17	4.4706e−02	7.3279e−02
SDIRK $p=2,\ s=3$	1.4989e+00	1.3883e+01	2.7766e+01	5.8955e+00	12	4.2222e−02	7.2031e−02
SDIRK $p=2,\ s=4$	1.4989e+00	1.3883e+01	2.7766e+01	5.8953e+00	9	4.2222e−02	7.2031e−02
SDIRK $p=2,\ s=5$	1.4983e+00	1.3767e+01	2.7534e+01	6.8196e+00	7	4.3429e−02	7.2637e−02
SDIRK $p=2,\ s=6$	1.4989e+00	1.3883e+01	2.7766e+01	5.8959e+00	6	4.2222e−02	7.2031e−02
SDIRK $p=3,\ s=2$	1.4201e+00	6.5856e+00	1.3171e+01	5.6088e+00	12	7.3131e−02	2.0743e−01
SDIRK $p=3,\ s=3$	1.3758e+00	5.0643e+00	1.0129e+01	4.1560e+00	9	5.9711e−02	2.3835e−01
SDIRK $p=3,\ s=4$	1.3273e+00	4.0259e+00	8.0517e+00	4.5424e+00	7	5.6066e−02	2.8454e−01
SDIRK $p=3,\ s=5$	1.2899e+00	3.4660e+00	6.9321e+00	3.8927e+00	6	5.1711e−02	3.2093e−01
SDIRK $p=3,\ s=6$	1.2395e+00	2.9075e+00	5.8151e+00	4.5133e+00	5	5.1385e−02	3.7544e−01

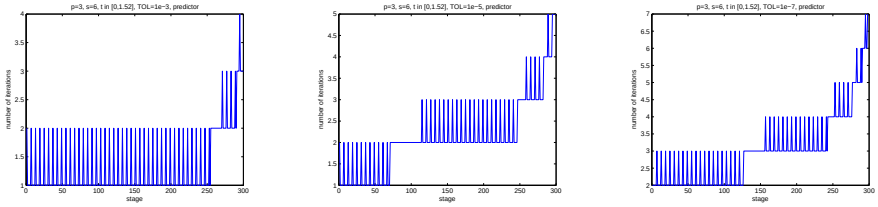
Table 2.8: Empirical radius of convergence of the fixed point iteration for (2.13), iteration tolerance 10^{-5}, maximal number of iterations = 100. First column: rightmost successful point t^*. Second column: solution value $\tan(t^*)$. Third column: Lipschitz constant $L = 2|\tan(t^*)|$. Fourth column: error at $t = 1.52$. Fifth column: Minimum number of gridpoints for convergence. Sixth column: Largest successful step. Seventh column: Predicted step ρ.



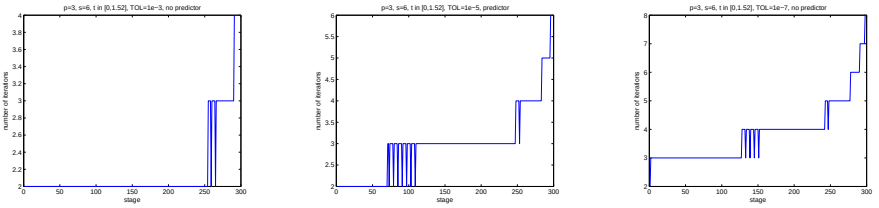
(a) TOL = 10^{-3} (407 iterations, with predictor)
(b) TOL = 10^{-5} (653 iterations, with predictor)
(c) TOL = 10^{-7} (920 iterations, with predictor)
(d) TOL = 10^{-3} (657 iterations, without predictor)
(e) TOL = 10^{-5} (903 iterations, without predictor)
(f) TOL = 10^{-7} (1170 iterations, without predictor)

Figure 2.2: Number of fixed point iterations with/without predictor, $p = 2,\ s = 6$

The observation of the error constants in Table 2.7 suggests to investigate the number of necessary fixed point iterations in dependence of the number of stages s. Table 2.9 gives the results for $p = 2, 3,\ s = 2, \dots 6$ for (2.13) on the interval $t \in [0, 1.52]$ with $N = 50$ steps. The entry for each pair (p, s) gives the total number of fixed point iterations necessary to compute the numerical approximation to the accuracy 10^{-9} in the left box and the average per stage, rounded to integers, in the right box. We observe that the average number of fixed point iterations per stage decreases for larger s and is also slightly smaller for the method with $p = 3$ than for $p = 2$.
This together with the error constants observed in Table 2.7 demonstrates that the increase in the number of stages has a beneficial effect on the accuracy and also on the convergence of the fixed point iteration which could balance the additional computational effort. In the next experiment, we investigate this aspect further:
For both the methods with $p = 2$ and $p = 3$, we compare the error and total number of fixed point iterations for the methods with $s = 2, \dots, 6$ applied to (2.13) with $t \in [0, 1.52]$. N is chosen such that the number of stages is equal in the integration



(a) TOL $= 10^{-3}$ (404 iterations, with predictor)
(b) TOL $= 10^{-5}$ (647 iterations, with predictor)
(c) TOL $= 10^{-7}$ (911 iterations, with predictor)



(d) TOL $= 10^{-3}$ (654 iterations, without predictor)
(e) TOL $= 10^{-5}$ (897 iterations, without predictor)
(f) TOL $= 10^{-7}$ (1161 iterations, without predictor)

Figure 2.3: Number of fixed point iterations with/without predictor, $p = 3,\ s = 6$

$p\backslash s$	$s=2$		$s=3$		$s=4$		$s=5$		$s=6$	
$p=2$	563	282	736	245	895	224	1050	210	1198	200
$p=3$	540	270	713	238	878	220	1035	207	1182	197

Table 2.9: Number of fixed point iterations for SDIRK 2 and SDIRK 3 with $s = 2, \ldots, 6$ applied to (2.13).

interval for all 6 methods, i.e. we choose $N = 300/s$. The results are given in Table 2.10. The left entry for each pair (p, s) gives the error at $t = 1.52$ and the right value is the total number of iterations required to reach the tolerance 10^{-9} in the fixed point iteration. We observe that for $p = 2$, the error of the solution obtained at the end of the integration is exactly the same for all s, but the number of fixed point iterations decreases slightly for higher s. For $p = 3$, the error is significantly smaller than for $p = 2$ at the same computational effort, again higher s requires a slightly reduced effort, but in this case the error increases with growing s.
Naturally, the performance of the SDIRK methods depends critically on the tolerance prescribed for the fixed point iteration. In Tables 2.11 and 2.12 we show the number of function evaluations necessary to achieve a global error smaller than 10^{-4}. For prescribed tolerances 'tol' for the fixed point iteration in $\{10^{-5}, 10^{-6}, 10^{-7}\}$, we list the step size Δt, error 'err' and number of evaluations of the right-hand side 'fcount'. For $p = 2$ and $s \leq 3$, the overall effort is smallest for the intermediate tolerance, and largest for the least stringent tolerance. For larger s, more stringent tolerances mean bigger computational effort. For $p = 3$, 'tol'$=10^{-5}$ generally yields the least efficient integration for all s, and the intermediate tolerance 10^{-6} gives the best results for $s \geq 3$. We conjecture that the reason for the bad performance with the least stringent tolerance are bad starting values for the fixed point iteration if the approximation is computed with low accuracy in the previous stage. Moreover, conversely to the case $p = 2$, for $p = 3$ the methods are more efficient for lower s.

$p\backslash s$	$s=2$		$s=3$		$s=4$		$s=5$		$s=6$	
$p=2$	4.4169e−02	1285	4.4169e−02	1241	4.4169e−02	1220	4.4169e−02	1207	4.4169e−02	1198
$p=3$	1.9217e−03	1252	2.5442e−03	1215	3.2474e−03	1199	3.9628e−03	1189	4.6734e−03	1182

Table 2.10: Error at $t = 1.52$ (left) and number of fixed point iterations (right) for SDIRK 2 and SDIRK 3 with $s = 2, \ldots, 6$ applied to (2.13), $N = 300/s$.

	Δt	err	fcount
SDIRK $p=2,\ s=2$, tol=10^{-5}	1.0397e−3	9.9684e−5	48160
SDIRK $p=2,\ s=2$, tol=10^{-6}	6.1538e−4	9.9892e−5	8626
SDIRK $p=2,\ s=2$, tol=10^{-7}	5.0381e−4	9.9937e−5	12084
SDIRK $p=2,\ s=3$, tol=10^{-5}	1.6187e−3	9.9900e−5	42080
SDIRK $p=2,\ s=3$, tol=10^{-6}	9.3194e−4	9.9930e−5	7796
SDIRK $p=2,\ s=3$, tol=10^{-7}	7.5547e−4	9.9950e−5	11204
SDIRK $p=2,\ s=4$, tol=10^{-5}	2.1902e−3	9.9180e−5	3934
SDIRK $p=2,\ s=4$, tol=10^{-6}	1.2490e−3	9.9990e−5	7383
SDIRK $p=2,\ s=4$, tol=10^{-7}	1.0073e−3	9.9999e−5	10763
SDIRK $p=2,\ s=5$, tol=10^{-5}	2.7636e−3	9.8600e−5	3770
SDIRK $p=2,\ s=5$, tol=10^{-6}	1.5638e−3	9.9788e−5	7148
SDIRK $p=2,\ s=5$, tol=10^{-7}	1.2583e−3	9.9881e−5	10504
SDIRK $p=2,\ s=6$, tol=10^{-5}	3.3407e−3	9.8730e−5	3659
SDIRK $p=2,\ s=6$, tol=10^{-6}	1.8812e−3	9.9857e−5	6982
SDIRK $p=2,\ s=6$, tol=10^{-7}	1.5094e−3	9.9830e−5	10330

Table 2.11: Step size Δt, error at $t = 1.52$ and number of evaluations of the right-hand side, comparison of different tolerances 'tol' for the fixed point iteration, for SDIRK with $p = 2$ applied to (2.13).

Additionally, a similar comparison with the Osher/Shu methods is given in Table 4.16 in Chapter 4.

2.3 Implementation of SDIRK methods in ANTARES

Memory–efficient realization of SDIRK methods

To save memory requirements for our algorithms, we state some pseudocode how to realize one step of the methods $y_{\text{old}} \to y_{\text{new}}$ for the differential equation $y'(t) = F(y(t))$ with only a total of two respectively three vectors in memory at the same time (plus one additionally for the fixed point iteration) for the second and third order SDIRK methods. The starting value for the fixed point iteration in each stage is given by a predictor computed by one step of the forward Euler method, which can be realized without an additional evaluation of the right-hand side.

SDIRK 2

```
yinc := yold;
for j=1..s do
%  solve ystage = yinc + h/2s * F(ystage);
  do until convergence
    ystage := yinc + h/2s * F(yold);
    yold := ystage;
  end do
  yold := 3*ystage - 2*yinc;   % = ystage + h/s F(ystage);
  yinc := 2*ystage - yinc;     % = ystage + h/2s F(ystage);
end j
ynew := yinc;
```

SDIRK 3

```
a := 0.5 * (1-sqrt((s-1)/(s+1)));
d := 1 / sqrt(s**2-1);
```

	Δt	err	fcount
SDIRK $p = 3,\ s = 2$, tol=10^{-5}	3.5415e−4	9.9972e−5	13291
SDIRK $p = 3,\ s = 2$, tol=10^{-6}	4.3182e−3	9.9005e−5	1729
SDIRK $p = 3,\ s = 2$, tol=10^{-7}	3.8287e−3	9.9916e−5	2210
SDIRK $p = 3,\ s = 3$, tol=10^{-5}	4.9625e−4	9.9994e−5	12700
SDIRK $p = 3,\ s = 3$, tol=10^{-6}	5.8915e−3	9.9549e−5	1765
SDIRK $p = 3,\ s = 3$, tol=10^{-7}	5.2234e−3	9.9736e−5	2274
SDIRK $p = 3,\ s = 4$, tol=10^{-5}	6.3758e−4	9.9963e−5	12386
SDIRK $p = 3,\ s = 4$, tol=10^{-6}	7.2381e−3	9.9372e−5	1829
SDIRK $p = 3,\ s = 4$, tol=10^{-7}	6.3866e−3	9.8791e−5	2385
SDIRK $p = 3,\ s = 5$, tol=10^{-5}	7.7909e−4	9.9938e−5	12184
SDIRK $p = 3,\ s = 5$, tol=10^{-6}	8.4444e−3	9.8348e−5	1898
SDIRK $p = 3,\ s = 5$, tol=10^{-7}	7.4510e−3	9.8773e−5	2487
SDIRK $p = 3,\ s = 6$, tol=10^{-5}	9.2065e−4	9.9974e−5	12036
SDIRK $p = 3,\ s = 6$, tol=10^{-6}	9.6203e−3	9.9184e−5	1950
SDIRK $p = 3,\ s = 6$, tol=10^{-7}	8.4444e−3	9.9054e−5	2583

Table 2.12: Step size Δt, error at $t = 1.52$ and number of evaluations of the right-hand side, comparison of different tolerances 'tol' for the fixed point iteration, for SDIRK with $p = 2$ applied to (2.13).

```
b := 1/s;

yinc := yold;
ysum := yold;
for j=1..s do
%  solve ystage = yinc + a*h * F(ystage);
  do until convergence
    ystage := yinc + a*h * F(yold);
    yold := ystage;
  end do
  ysum := ysum + (b/a) * (ystage-yinc);     % = ysum + b*h * F(ystage);
  yold := ystage + (d/a) * (ystage-yinc);   % = ystage + d*h * F(ystage);
  yinc := yinc + (d/a) * (ystage-yinc);     % = ystage + d*h * F(ystage);
end j
ynew := ysum;
```

For a simple forward integration, the memory requirement of this realization seems optimal. In the next section, we describe an adaptive step-size selection algorithm which additionally makes it necessary to store the variable `yold`. Under these circumstances, we can reduce the memory requirement by the following modification:

```
a := 0.5 * (1-sqrt((s-1)/(s+1)));
d := 1 / sqrt(s**2-1);
b := 1/s;

ynought := yold;
yinc := yold;
for j=1..s do
%  solve ystage = yinc + a*h * F(ystage);
  do until convergence
    ystage := yinc + a*h * F(yold);
    yold := ystage;
```

```
  end do
  yold := ystage + (d/a) * (ystage-yinc);   % = ystage + d*h * F(ystage);
  yinc := yinc + (d/a) * (ystage-yinc);     % = yinc + d*h * F(ystage);
end j
ynew := ynought + (b/d) * (yinc - ynought);
```

Selection of time-steps

According to (2.4), the time-step has to be reduced when the fixed point iteration to compute the SDIRK approximation does not converge. According to the results in Table 2.10, an increase in the number of stages s might also improve convergence. Accommodating for these observations, we have implemented the following heuristics:

```
Choose initial CFL-number C;
Calculate h0;
Choose smin;
Choose smax;
Choose initial s in smin..smax;
itn_exp := 15;      % accept the fixed point iteration without convergence
itn_chg := 10;      % reduce stepsize if no convergence
succ_itn := 50;     % after succ_itn successful steps, increase stepsize
min_cour := 0.1;    % lowest acceptable courant number
cour_inc := 1.25;   % factor to increase stepsize
cour_dec := 2/3;    % factor to decrease stepsize
h := h0*C;

for all timesteps do
  Calculate h0;                 % Spiegel formula and other controls
  h := h0*C;
  if (succ_itn successful time steps with C in galaxy 1) then
  % 'galaxy' refers to one particular of the nested grids
    if (C >= 0.5*s) then
      s := max( smin , s-2 );
    end if
    C := min( cour_inc*C , 0.5*s );
    h := h0*C;
  end if
  if ('no convergence' or itn_chg steps) then
    C := max(min_cour , cour_dec*C);
    h := h0*C;
    if (C <= min_cour) then
      if (s >= smax) then
        if (# iteration steps < itn_exp) then
          do another iteration;
        else
          accept solution       % corresponds to a step of an explicit Runge-Kutta method
        end if
      else
        s := min( s+2 , smax );
      end if
    end if
  end if
end do
```

`'no convergence'` is assumed if the relative change in the fixed point iteration error drops below `rel_tol`:=10^{-3}. Conversely, the iterate is accepted if the relative error is less than `req_err`=10^{-5}.
The relative change between iterates x^o and x^n is computed as follows for the state variable $x \in \{\rho, e, m_x, m_y, m_z\}$:

- Let $x = (x_{j,k}) \in \{\rho, e\}$ be either the density or the energy. Let $j \in L := \{1, \ldots, J\}$ refer to the index of a horizontal layer which itself corresponds to an index set $L_j := \{1, \ldots, K_j\},\ K_j \equiv K$, and define

$$N_j := \max_{k \in L_j} \max\{|x^o_{j,k}|, |x^n_{j,k}|\} + R,$$

  where $R := 10^{-11}$ for $x = \rho$ and $R := 1$ for $x = e$. Then

$$\|x^o - x^n\| := \max_{j \in L} \frac{1}{N_j} \left(\sum_{i \in L_j} |x^o_{j,i} - x^n_{j,i}|^2 \right)^{1/2}. \tag{2.20}$$

  Alternatively, the norm $\max_{j \in L}$ could be replaced by any vector norm.

- Let $x \in \{m_x, m_y, m_z\}$ be a component of the momentum. Note that in the current 2D implementation, m_z is void. Let c_s refer to the (local) velocity of sound (again with upper indices to indicate the iteration step), which has the dimension of an $L \times K$ matrix and introduce a parameter $q := 1$. Then in (2.20) we replace N_j by

$$N_j := \max_{k \in L_j} \max\{|x^o_{j,k}|, |x^n_{j,k}|, q(c^n_s)_{j,k}\rho^o_{j,k}, q(c^n_s)_{j,k}\rho^n_{j,k}\}. \tag{2.21}$$

Implementation in ANTARES

For convenience of usage of the SDIRK implementation in the ANTARES code, we list the most important changes and parameters. Our explanations refer to the pseudocode in the last paragraphs. In ANTARES, the variables are associated with the pseudocode as follows:

u0(:,:,:,:,l_u0_p2) ... `ynought`
u0(:,:,:,:,l_u0_p1) ... `yinc`
u0(:,:,:,:,l_u0_o) ... `yold`
u0(:,:,:,:,l_u0_n) ... `ystage`

- `input.a`:
  l.13: MET_RK = 3 ! calls the SDIRK method of second order, MET_RK = 4 calls SDIRK 3.
  l.14: SDIRK_S ! sets the initial SDIRK_S.
  l.14: SDIRK_S_MIN ! sets the lower limit `smin` on the number of stages SDIRK_S.
  l.14: SDIRK_S_MAX ! sets the upper limit `smax` on the number of stages SDIRK_S.
  l.21: C ! sets the initial CFL number `C`.
  l.21: tau_max ! defines an upper limit on the step size.

- `1b_g.f90`:
  itn_exp = 15 ! accept the fixed point iteration without convergence.
  itn_chg = 10 ! reduce stepsize if no convergence.
  succ_itn = 5 ! after succ_itn successful steps in galaxy 1, increase stepsize.
  req_err = 1.d-3 ! relative error tolerance for fixed point iteration.
  rel_tol = 1.d-3 ! relative change in iteration errors to assume `'no convergence'`
  min_cour = 1.d-1 ! sets the lowest acceptable courant number.
  cour_dec = 2.d0/3.d0 ! sets the increase factor for the courant number.
  cour_inc = 1.25 ! sets the decrease factor for the courant number.

- `1i_u.f90`:
  `subroutine TAU_SUN`: maximal admissible Courant number is defined as 0.5*SDIRK_S.
  be_rk ! RK coefficients defined for SDIRK, be_rk = a_{ii}.
  al_rk ! RK coefficients defined for SDIRK, al_rk = (0,1,0) (multiplies `yinc` by 1).

- `1h.f90`:
  `function CHECK_IT`: stepsize is reduced if necessary and predictor is evaluated.
  `function CALC_IT_ERR`: the current iteration error is evaluated.
  `function DO_STG`: initialization `yinc := yold`.
  `function DO_STG`: initial value `ynought` is saved for adaptive time-stepping.

Remark: Note that the parameter `req_err` is a (relative) tolerance for successful termination of the fixed point iteration, while `rel_tol` represents the minimum required change in the improvement of the approximation to continue the iteration. Thus, `req_err` refers to the change in the iterate, `rel_tol` to the change in `req_err`.

lines 13 and 14 in `input.a` should read as follows to set the parameters for the SDIRK methods:

```
3       MET_RK: 0: Euler forward, 1: TVD/2, 2: TVD/3, 3: SDIRK/2, 4: SDIRK/3
6 6 6   SDIRK_S, SDIRK_S_MIN, SDIRK_S_MAX
```

Additional output is created to monitor the evolution of the step-sizes and number of fixed point iterations:
The file `fort.800` contains the following columns:

universe(1)%tot_scrt	...	absolute time in units 'sound-crossing times'.
universe(1)%stp	...	step index in galaxy 1.
stp	...	step index in current galaxy.
stg	...	stage index in current galaxy.
act_ind_gal	...	index of current galaxy.
universe(1)%cour_no	...	courant number in galaxy 1.
sdirk_s	...	number of stages for SDIRK.
i	...	iteration index.

This output is generated when the CFL number `C` or the number of stages `s` changes, and additionally for each step where output is written. The output generates two lines with the old and the new values, respectively.
The file `fort.801` contains the following columns:

universe(1)%tot_scrt	...	absolute time in units 'sound-crossing times'.
act_ind_gal	...	index of current galaxy.
stp	...	step index in current galaxy.
stg	...	stage index in current galaxy.
i	...	iteration index.

This output is generated after each stage.

Chapter 3

Dissipativity analysis

In this chapter, we analyze the *dissipativity* of spatial discretisations and time integrators implemented in the ANTARES code. Dissipative numerical schemes have the property of damping out high-frequency waves that can make the computed solution more oscillatory than desired or physically plausible, and which make time integration unstable. The requirement of this property commonly implies a restriction on the admissible time step-size, even for appropriate space discretisations.

3.1 Construction of dissipative spatial discretisations

A simple necessary condition to test whether a given spatial discretisation in a numerical scheme for the solution of a parabolic partial differential equation (PDE) is dissipative is the following:
Define the oscillatory function ϵ defined on a grid of points $x_i,\ i \in \mathbb{D} \subseteq \mathbb{N}$:

$$\begin{aligned} \epsilon(x_i) &:= +1 \quad \text{for some fixed choice of } i, \\ \epsilon(x_{i\pm k}) &:= -1 \quad \text{for all odd } k, \\ \epsilon(x_{i\pm k}) &:= +1 \quad \text{for all even } k, \end{aligned} \tag{3.1}$$

i.e. the sequence of numbers $\{..., +1, -1, +1, -1, +1, -1, ...\}$ which we choose as an initial condition $u_0 = u(x, t_0) = \epsilon(x)$ at the points $x = x_i$ for the heat equation

$$u_t = b\, u_{xx}, \tag{3.2}$$

where b is a constant coefficient. The grid points x_i are considered here to be equi-spaced and located at the centers of the grid cells, the function $\epsilon(x)$ hence is constant within each grid cell. In the simplest case the time integration can be realized by the forward Euler scheme such that

$$U_i^{n+1} = U_i^n + (b\,\Delta t)\,(U_i^n)_{xx} \tag{3.3}$$

for a numerical approximation $U_i^n \approx u(x_i, t_n)$. The exact solution of (3.2) satisfies $\lim_{t\to\infty} u(x,t) = 0$ monotonically as $t \to \infty$ [44, 46]. Hence, any discretisation for $(U_i^n)_{xx}$ should have the same property and must thus satisfy $-2 < b\Delta t (U_i^n)_{xx} < 0$ for damping and $-1 < b\Delta t (U_i^n)_{xx} < 0$ for monotonic damping. An equivalent requirement is to consider the Fourier transform of the discretisation and require that the amplification factor satisfies $|g(\theta)| < 1$ for the maximum values of $|\theta|$, i.e. for $\theta = \pm\pi$, see Section 3.3 for more details. In conjunction with an appropriate time integrator (time discretisation) such schemes will usually be dissipative. We now discuss several schemes which have been or could be considered to be used for parabolic terms in the fully compressible Navier-Stokes equations and their associated energy equation for use in conservative finite difference / finite volume based solvers such as those employed within the ANTARES code [32].

3.2 Standard schemes and properties required for dissipativity

The standard scheme to approximate $(U_i^n)_{xx}$ to second order is the centred 3-point stencil $(1, -2, 1)$, i.e.

$$(U_i^n)_{xx} = \frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{(\Delta x)^2} + O((\Delta x)^2). \tag{3.4}$$

To first order in Δx, (3.3) with (3.4) yields

$$U_i^{n+1} = U_i^n \left(1 - 4b\,\frac{\Delta t}{(\Delta x)^2}\right) \tag{3.5}$$

for $n \geq 0$ with $u_0 = \epsilon(x)$. The scheme is dissipative for

$$\Delta t < \frac{1}{2b}(\Delta x)^2, \tag{3.6}$$

which is no additional restriction to the stability constraint (stability holds for $\Delta t = 0.5(\Delta x)^2/b$ as well, but this would not preserve the long-term behaviour of the exact solution) and

$$\Delta t < \frac{1}{4b}(\Delta x)^2 \tag{3.7}$$

for monotonic decay of the numerical approximation.
Remarkably, the 3-point stencil $(1, -2, 1)$ leads to M-matrix properties [33, p. 29] of the associated differentiation matrix, which is obtained from (3.4) on a finite grid. To be more precise, the matrix is connected (thus the solution is not evolved on disjoint subgrids), it is (weakly) diagonally dominant with the boundary conditions ensuring that the strict inequality holds for at least one row of the matrix, and the diagonal entries are positive and off-diagonal entries non-positive.
It can easily be seen that the M-matrix property guarantees dissipativity: since the diagonal dominates, the mid-point of the stencil must be strictly negative to yield any damping at all. Since the constant function $u(x,t) = c$ must also be a valid initial condition of (3.2), the sum of the coefficients of the stencil must be zero, as required by algebraic conditions for the consistency of the discretisation, hence strict diagonal dominance is not permitted except for the boundary conditions. Thus, the off-center entries of the stencil must all be non-negative, and in general they also have to be positive to avoid disconnectedness. The stencil $(1, -2, 1)$ satisfies all these conditions. However, in general only pseudospectral methods will lead to such stencils, since flexible collocation is required to generate positive off-center contributions. This is not the case for polynomial based (finite order) stencils on uniform grids. Since uniform or logarithmically graded grids are required for ENO interpolations, it would be convenient if the M-matrix property were not needed.

3.2.1 Connectedness

We now introduce a second order scheme constructed similarly to the one used in ANTARES [32]. We assume that

$$(U_i^n)_x = \frac{U_{i+1}^n - U_{i-1}^n}{2\Delta x} + O((\Delta x)^2), \tag{3.8}$$

and with

$$(\hat{f}_{i+1/2})^n = \frac{1}{2}\left((U_i)_x + (U_{i+1})_x\right) + O((\Delta x)^2) = \frac{U_{i+2}^n + U_{i+1}^n - U_i^n - U_{i-1}^n}{4\Delta x} \tag{3.9}$$

and

$$(\hat{f}_{i-1/2})^n = \frac{1}{2}\left((U_{i-1})_x + (U_i)_x\right) + O((\Delta x)^2) = \frac{U_{i+1}^n + U_i^n - U_{i-1}^n - U_{i-2}^n}{4\Delta x}, \tag{3.10}$$

we can compute

$$(U_i^n)_{xx} = \mathrm{div} f_i^n = \frac{(\hat{f}_{i+1/2})^n - (\hat{f}_{i-1/2})^n}{\Delta x} + O((\Delta x)^2) = \frac{U_{i+2}^n - 2U_i^n + U_{i-2}^n}{(2\Delta x)^2}, \tag{3.11}$$

which is the 5-point stencil (1,0,-2,0,1). It is weakly diagonally dominant, but not connected (the solution is obtained on two separate grids defined for odd and even values of i). If we use this in (3.2) with initial data ϵ from (3.1), we obtain $(U_i^n)_{xx} = 0$ and thus the resulting scheme is stable, but (strictly) non-dissipative for any positive Δt.
This consideration demonstrates that weak diagonal dominance is not sufficient for dissipativity. In the next section we will demonstrate that this property is also not necessary for a scheme to be dissipative.

3.2.2 A non-diagonally dominant, dissipative scheme

The following scheme will demonstrate that the requirement of diagonal dominance of the differentiation matrix (and the stencil) is not even necessary for a scheme to be dissipative. Consider the fourth order approximation

$$(U_i^n)_{xx} = \frac{-U_{i+2}^n + 16\,U_{i+1}^n - 30\,U_i^n + 16\,U_{i-1}^n - U_{i-2}^n}{12\,(\Delta x)^2} + O((\Delta x)^4), \tag{3.12}$$

which is obtained following [43] (Exercise 6.3.12b, right-hand side of sample equation) or directly from Table 3.1-1 in [9]. This is a 5-point stencil given by $(-1, +16, -30, +16, -1)/12$. It is connected, but not diagonally dominant. Similarly to (3.5) we obtain

$$U_i^{n+1} = U_i^n \left(1 - 16\,b\,\frac{\Delta t}{3(\Delta x)^2)}\right) \tag{3.13}$$

for $n = 0$ with $u_0 = \epsilon(x)$. The scheme is dissipative for the initial condition given by the function $\epsilon(x)$ from (3.1). Indeed, this function leads to all negative entries for

$$\Delta t < \frac{3}{8\,b}(\Delta x)^2, \tag{3.14}$$

and if even

$$\Delta t < \frac{3}{16\,b}(\Delta x)^2, \tag{3.15}$$

the numerical approximation behaves monotonically. Hence, the M-matrix property is not required for a scheme to be dissipative. The dissipativity analysis for this space discretisation is performed in Section 3.3.4 for the SDIRK methods and in Section 4 for the Osher/Shu methods.
In Section 3.2.6 below, we will show that this scheme can be derived within the framework implemented in the ANTARES code, if interpolation of first derivatives is performed on the cell boundaries instead of the cell centers in the derivation of the scheme, while still imposing a conservation constraint (3.28).

3.2.3 Alternating sign property of connected stencils for dissipative schemes

Following the above example (3.12), it is straightforward to see that a centred stencil with an odd number of points, which is algebraically consistent (the sum of the weights equals zero, whence the constant function is integrated correctly in time to remain constant, if given as initial data for (3.2) on the torus), connected (no inner weights equal zero), has negative weight at the main diagonal (center of the stencil) and *alternating signs* for all the other non-zero elements of the stencil, is dissipative for (3.1)–(3.3). The function $\epsilon(x)$ from (3.1) then leads to all negative entries for $(U_i^n)_{xx}$, which in turn implies more stringent restrictions in the time step size Δt as compared to stencils with the M-matrix property, but the same holds for stability in general for this class of stencils. For the forward Euler integration the resulting scheme can always be made dissipative (and stable) for small enough Δt.
As a result, for (3.1)–(3.3) all stencils based on centred polynomial interpolation on equidistant grids for second spatial derivatives (with a resulting odd number of nodes) lead to dissipative schemes of this class, as can also be seen from the examples of up to 8^{th} order in Table 3.1-1 and the exact expressions following (3.2-2) in Chap. 3.2 of [9] and references therein. The approximation (3.12) is just the special case for fourth order. The classical 3-point centred stencil which leads to (3.4) is contained in this class as another special case and as the only case which belongs to both stencils with M-matrix property and to connected stencils with alternating sign property.

3.2.4 Non-dissipative schemes in conservative hydro-solvers

The following schemes have been used in conservative, numerical simulation codes and can be shown to be non-dissipative. They satisfy the properties of flux conservation, easy generalization from 1D to an arbitrary number of dimensions through directional (tensor product) splitting, and easy implementation into conservative numerical schemes for the hyperbolic part of the hydrodynamical equations, where parabolic terms appear due to transport processes on a microscopic length scale (heat and momentum diffusion).

They are obtained by computing first inner derivatives at the grid center using a standard, antisymmetric 5-point stencil (Table 3.1-1 of [9]):

$$(U_i^n)_x = \frac{-U_{i+2}^n + 8\,U_{i+1}^n - 8\,U_{i-1}^n + U_{i-2}^n}{12\,h} + O(h^4). \tag{3.16}$$

As required by anti-symmetry for first order derivative approximations the central weight is zero in this case. In a second stage, fluxes $(\hat{f}_{i+1/2})^n$ and $(\hat{f}_{i-1/2})^n$ are computed from a second, centred interpolation to the boundaries of the cell around the grid point x_i. Straightforward interpolation yields the weights $(-1/16, 9/16, 9/16, -1/16)$ as used in ANTARES [32], while interpolation of first derivatives on the cell boundaries assuming a conservation constraint (3.28) yields the weights $(-1/12, 7/12, 7/12, -1/12)$ for the set of grid points $x_{i-2}, x_{i-1}, x_i, x_{i+1}$ and $x_{i-1}, x_i, x_{i+1}, x_{i+2}$ on which derivatives $\partial u/\partial x$ have to be computed by means of (3.16) to obtain $(\hat{f}_{i+1/2})^n$ and $(\hat{f}_{i-1/2})^n$ from interpolation. Similarly to (3.11) we then obtain

$$(U_i^n)_{xx} = \frac{U_{i+4}^n - 18\,U_{i+3}^n + 80\,U_{i+2}^n + 18\,U_{i+1}^n - 162\,U_i^n + 18\,U_{i-1}^n + 80\,U_{i-2}^n - 18\,U_{i-3}^n + U_{i-4}^n}{192\,h^2}, \tag{3.17}$$

for the approximation used in [32] and the alternative approximation

$$(U_i^n)_{xx} = \frac{U_{i+4}^n - 16\,U_{i+3}^n + 64\,U_{i+2}^n + 16\,U_{i+1}^n - 130\,U_i^n + 16\,U_{i-1}^n + 64\,U_{i-2}^n - 16\,U_{i-3}^n + U_{i-4}^n}{144\,h^2}. \tag{3.18}$$

Both stencils are connected, but are neither diagonally dominant nor do they satisfy the alternating sign property. Rather, for the function $\epsilon(x)$ from (3.1) we obtain that some contributions cancel each other (all j such that $|j-i|$ is an odd integer) while the others together cancel the contribution from the main diagonal (all j such that $|j-i|$ is an even integer), whence $(U_i^n)_{xx} = 0$ and the two approximations (3.17) and (3.18) are non-dissipative on the grid scale, as has also been confirmed by a complete dissipativity analysis based on Fourier methods for various Runge-Kutta schemes in Section 3.3 (there, $g(\pi) = g(-\pi) = 1$ contains the special case of using $\epsilon(x)$ from (3.1) as an initial condition, where g denotes the amplification factor defined in Section 3.3 below).

3.2.5 Staggered mesh approach to derive dissipative schemes

One can assume a staggered mesh point of view to directly compute the fluxes which appear inside the outer (divergence-like) derivative of the parabolic terms at the cell boundary. Thus,

$$(\hat{f}_{i+1/2})^n = b_{i+1/2}\frac{U_{i+1}^n - U_i^n}{h} + O(h^2) \tag{3.19}$$

and

$$(\hat{f}_{i-1/2})^n = b_{i-1/2}\frac{U_i^n - U_{i-1}^n}{h} + O(h^2), \tag{3.20}$$

which for constant b yields

$$b\,(U_i^n)_{xx} = b\,\frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{h^2} + O(h^3). \tag{3.21}$$

This corresponds with (3.4), which has been shown to have all the desired properties.
We recall that the procedure used to derive (3.17) and (3.18) when applied to second derivatives gives the non-connected stencil of the approximation (3.11). All three schemes are non-dissipative. This suggests the following, alternative procedure to derive a new replacement for (3.17) and (3.18) by following a staggered mesh approach when using fourth order approximations. Following [9, Table on p. 91 in Chap. 5.3], the staggered, centred fourth order approximation of first derivatives, which here we take to be the fluxes (in general they would have to be multiplied with interpolants for the diffusivities), are

$$(\hat{f}_{i+1/2})^n = \frac{8U_{i-1}^n - 9\,U_i^n + 9\,U_{i+1}^n - 8U_{i+2}^n}{192\,h} + O(h^4) \tag{3.22}$$

and

$$(\hat{f}_{i-1/2})^n = \frac{8U_{i-2}^n - 9\,U_{i-1}^n + 9\,U_i^n - 8U_{i+1}^n}{192\,h} + O(h^4), \tag{3.23}$$

whence

$$(U_i^n)_{xx} \approx \mathrm{div} f_i^n = \frac{(\hat{f}_{i+1/2})^n - (\hat{f}_{i-1/2})^n}{h} = \frac{-U_{i+2}^n + 28\,U_{i+1}^n - 54\,U_i^n + 28\,U_{i-1}^n - U_{i-2}^n}{24\,h^2}. \tag{3.24}$$

Unfortunately, it turns out that this scheme does not have the full fourth order. Thus, a modification is necessary. Full order can be retained if, alternatively to the previous interpolation of the cell centres underlying (3.22)–(3.23), the cell boundaries are interpolated in such a way that the volume integrals are exact. We investigate a scheme constructed by this principle in Section 3.3.4 below. This approximation is based on a connected stencil which has the alternating sign property and thus is quickly shown to be dissipative for (3.1)–(3.3). The time step limits are only slightly more strict than for (3.4) and less strict than for (3.12) and can be summarised as follows. It can be shown that

$$U_i^{n+1} = U_i^n \left(1 - \frac{14\,b}{3}\,\frac{\Delta t}{(\Delta x)^2}\right) \tag{3.25}$$

for $n = 0$ with $u_0 = \epsilon(x)$. The scheme is dissipative for

$$\Delta t < \frac{3}{7b}(\Delta x)^2 \tag{3.26}$$

and if even

$$\Delta t < \frac{3}{14b}(\Delta x)^2, \tag{3.27}$$

then the behavior of the scheme is strictly monotonic. We note that the coefficients of (3.24) differ from (3.12), particularly, the weights of the outermost points of the stencil are just one half of it. This concentration of weights is typical for the staggered mesh procedure, but by uniqueness already it is clear that the spatial discretisation in (3.24) cannot be of fourth order. The scheme has to be further modified to guarantee fourth order, or conservation of fluxes. We further note that there is an interchange of the sequence of interpolation of diffusion coefficients and multiplication with the first, inner derivative as compared to the current procedure which yields (3.17) or (3.18), but otherwise it satisfies all the properties required for a convenient application in conservative, high order numerical hydrodynamical simulation codes.

3.2.6 Schemes with interpolation of cell boundaries

In the ENO-approach [36], U_i are assumed to be cell averages of a function $v(x)$,

$$U_i = \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} v(\zeta)\,d\zeta. \tag{3.28}$$

Approximation of $v(x)$ by a polynomial function $p(x)$ to specified order, $p(x) = v(x) + O(h^5)$, yields

$$(U_i)_x = \frac{1}{h}(p(x_{i+1/2}) - p(x_{i-1/2})) + O(h^5) \tag{3.29}$$

as long as the functions are smooth enough to give an extra h in the difference to cancel the corresponding factor in the denominator. Clearly, using $p(x_{i+1/2})$ to approximate $U_{i+1/2}$ leads to a reduced order of accuracy, higher order can only result from cancellation in forming the difference. Using $p'(x_{i\pm1/2})$ instead of the $\hat{f}_{i\pm1/2}$ from the previous section for forming the second derivative we find

$$(U_i)_{xx} = \frac{p'(x_{i+1/2}) - p'(x_{i-1/2})}{h}. \tag{3.30}$$

A centred four point stencil approximation leads to

$$p'_{i+1/2} = \frac{U_{i-1} - 15\,U_i + 15\,U_{i+1} - U_{i+2}}{12\,h} + O(h^2), \tag{3.31}$$

and

$$p'_{i-1/2} = \frac{U_{i-2} - 15\,U_{i-1} + 15\,U_i - U_{i+1}}{12\,h} + O(h^2), \tag{3.32}$$

θ	$g(\theta)$
$-\pi$	$-\frac{-1+2\,\mu}{1+2\,\mu}$
$-\frac{3\pi}{4}$	$-\frac{-2+\mu\,\sqrt{2}+2\,\mu}{2+\mu\,\sqrt{2}+2\,\mu}$
$-\frac{\pi}{2}$	$-\frac{-1+\mu}{1+\mu}$
$-\frac{\pi}{4}$	$-\frac{2+\mu\,\sqrt{2}-2\,\mu}{-2+\mu\,\sqrt{2}-2\,\mu}$
0	1
$\frac{\pi}{4}$	$-\frac{2+\mu\,\sqrt{2}-2\,\mu}{-2+\mu\,\sqrt{2}-2\,\mu}$
$\frac{\pi}{2}$	$-\frac{-1+\mu}{1+\mu}$
$\frac{3\pi}{4}$	$-\frac{-2+\mu\,\sqrt{2}+2\,\mu}{2+\mu\,\sqrt{2}+2\,\mu}$
π	$-\frac{-1+2\,\mu}{1+2\,\mu}$

Table 3.1: Values of $g(\theta)$ for some θ, $p=2,\ s=1$.

where the order of accuracy is determined with respect to $(U_{i\pm1/2})_x$. This has been calculated with a MATHEMATICA 4.1™ script. Accordingly, the second derivative is given by

$$(U_i^n)_{xx} = \frac{-U_{i-2} + 16\,U_{i-1} - 30\,U_i + 16\,U_{i+1} - U_{i+2}}{12\,h^2} + O(h^4), \tag{3.33}$$

which corresponds to the fourth order approximation (3.12).

3.3 Investigation of the spatial discretisations

In addition, we have also performed the dissipativity analysis of space and time discretisations in our focus as in [43] for the heat equation (3.2).
The crucial quantity that we study in this context is the *amplification factor* $g = g(\theta)$: the propagation of the numerical solution by one time step corresponds to the multiplication of the numerical solution's Fourier transform by this factor g [43, Section 2.2]. Thus, the magnitude of the amplification factor shows the factor by which the amplitude of each frequency in the (spatial approximation of the) solution is increased/decreased in each time step.

3.3.1 Three-point difference scheme

The corresponding analysis has been performed for the three-point difference scheme (3.21) in space with the SDIRK methods as time integrators, with $s = 1, \ldots, 4$ for both the schemes with $p = 2$ and $p = 3$. For $\mu = b\frac{\Delta t}{(\Delta x)^2}$ we analyze the relationship

$$g(\theta) = R\left(\mu\left(\mathrm{e}^{\mathrm{i}\theta} - 2 + \mathrm{e}^{-\mathrm{i}\theta}\right)\right), \tag{3.34}$$

which is the extension of [43, p.146].
We obtained the following results:
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s = 1$:

$$g(\theta) = -\frac{\mu\,\cos(\theta) + 1 - \mu}{\mu\,\cos(\theta) - 1 - \mu}. \tag{3.35}$$

In Table 3.1, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = 1$ and $|g(\theta)| < 1,\ \theta \neq 0$.
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s = 2$:

$$g(\theta) = \frac{\mu^2\,(\cos(\theta))^2 - 2\,\mu^2\cos(\theta) + 4\,\mu\,\cos(\theta) + \mu^2 - 4\,\mu + 4}{\mu^2\,(\cos(\theta))^2 - 4\,\mu\,\cos(\theta) - 2\,\mu^2\cos(\theta) + 4 + 4\,\mu + \mu^2}. \tag{3.36}$$

θ	$g(\theta)$
$-\pi$	$\frac{(\mu-1)^2}{(\mu+1)^2}$
$-\frac{3\pi}{4}$	$\frac{\left(\mu\sqrt{2}+2\,\mu-4\right)^2}{\left(4+\mu\sqrt{2}+2\,\mu\right)^2}$
$-\frac{\pi}{2}$	$\frac{(\mu-2)^2}{(2+\mu)^2}$
$-\frac{\pi}{4}$	$\frac{\left(\mu\sqrt{2}-2\,\mu+4\right)^2}{\left(-4+\mu\sqrt{2}-2\,\mu\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(\mu\sqrt{2}-2\,\mu+4\right)^2}{\left(-4+\mu\sqrt{2}-2\,\mu\right)^2}$
$\frac{\pi}{2}$	$\frac{(\mu-2)^2}{(2+\mu)^2}$
$\frac{3\pi}{4}$	$\frac{\left(\mu\sqrt{2}+2\,\mu-4\right)^2}{\left(4+\mu\sqrt{2}+2\,\mu\right)^2}$
π	$\frac{(\mu-1)^2}{(\mu+1)^2}$

Table 3.2: Values of $g(\theta)$ for some θ, $p=2,\ s=2$.

In Table 3.2, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi+\ell\pi/4 : \ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu>0$, with $g(0)=1$ and $|g(\theta)|<1,\ \theta\neq 0$.
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s=3$:

$$g(\theta) = -\frac{\mu^3\left(\cos(\theta)\right)^3 - 3\,\mu^3\left(\cos(\theta)\right)^2 + 9\,\mu^2\left(\cos(\theta)\right)^2 + 3\,\mu^3\cos(\theta) - 18\,\mu^2\cos(\theta) + 27\,\mu\cos(\theta) - \mu^3 + 9\,\mu^2 - 27\,\mu + 27}{\mu^3\left(\cos(\theta)\right)^3 - 9\,\mu^2\left(\cos(\theta)\right)^2 - 3\,\mu^3\left(\cos(\theta)\right)^2 + 27\,\mu\cos(\theta) + 18\,\mu^2\cos(\theta) + 3\,\mu^3\cos(\theta) - 27 - 27\,\mu - 9\,\mu^2 - \mu^3}. \tag{3.37}$$

In Table 3.3, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi+\ell\pi/4 : \ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu>0$, with $g(0)=1$ and $|g(\theta)|<1,\ \theta\neq 0$.
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$. We do not reproduce the general expression for g here because it is too complicated for proper typesetting. In Table 3.4, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi+\ell\pi/4 : \ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu>0$, with $g(0)=1$ and $|g(\theta)|<1,\ \theta\neq 0$.
$\underline{p=3:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s=2$:

$$g(\theta) = -\frac{\left(1+\sqrt{3}\right)\left(4\mu^2\left(\cos(\theta)\right)^2 + 2\sqrt{3}\mu\cos(\theta) - 8\,\mu^2\cos(\theta) + 6\,\mu\cos(\theta) + 4\mu^2 + 3 + 3\sqrt{3} - 2\sqrt{3}\mu - 6\,\mu\right)}{-4\mu^2\left(\cos(\theta)\right)^2 + (12+4\sqrt{3})\,\mu\cos(\theta) + 8\,\mu^2\cos(\theta) - 12 - 12\,\mu - 6\sqrt{3} - 4\mu^2 - 4\sqrt{3}\,\mu}. \tag{3.38}$$

In Table 3.5, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi+\ell\pi/4 : \ell=0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)|<1$.
$\underline{p=3:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s=3$:

$$\begin{aligned}
g(\theta) &= \frac{P_1}{P_2}, \qquad (3.39)\\
P_1 &= \left(5+2\sqrt{2}\right)\Big((45\sqrt{2}+66)\mu^2\left(\cos(\theta)\right)^2 + 45\sqrt{2}\mu^2 - 51\,\mu^3\left(\cos(\theta)\right)^2 - 17\,\mu^3 + 66\mu^2\cdots\\
&\quad +17\,\mu^3\left(\cos(\theta)\right)^3 + 51\,\mu^3\cos(\theta) + (108\sqrt{2}+138)\mu\cos(\theta) - (90\sqrt{2}+132)\mu^2\cos(\theta) + \cdots\\
&\quad +132 + 90\sqrt{2} - (108\sqrt{2}+138)\,\mu\Big),\\
P_2 &= 51\Big(20 + 14\sqrt{2} + (12\sqrt{2}+18)\,\mu + (6+3\sqrt{2})\,\mu^2 + \mu^3 - (12\sqrt{2}+18)\,\mu\cos(\theta) + 3\,\mu^3\left(\cos(\theta)\right)^2 + \cdots\\
&\quad +(3\sqrt{2}+6)\mu^2\left(\cos(\theta)\right)^2 - \mu^3\left(\cos(\theta)\right)^3 - 3\,\mu^3\cos(\theta) - (6\sqrt{2}\mu^2+12)\cos(\theta)\Big).
\end{aligned}$$

θ	$g(\theta)$
$-\pi$	$-\frac{(2\,\mu-3)^3}{(3+2\,\mu)^3}$
$-\frac{3\pi}{4}$	$-\frac{\left(\mu\,\sqrt{2}+2\,\mu-6\right)^3}{\left(6+\mu\,\sqrt{2}+2\,\mu\right)^3}$
$-\frac{\pi}{2}$	$-\frac{(\mu-3)^3}{(3+\mu)^3}$
$-\frac{\pi}{4}$	$-\frac{\left(\mu\,\sqrt{2}-2\,\mu+6\right)^3}{\left(-6+\mu\,\sqrt{2}-2\,\mu\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{\left(\mu\,\sqrt{2}-2\,\mu+6\right)^3}{\left(-6+\mu\,\sqrt{2}-2\,\mu\right)^3}$
$\frac{\pi}{2}$	$-\frac{(\mu-3)^3}{(3+\mu)^3}$
$\frac{3\pi}{4}$	$-\frac{\left(\mu\,\sqrt{2}+2\,\mu-6\right)^3}{\left(6+\mu\,\sqrt{2}+2\,\mu\right)^3}$
π	$-\frac{(2\,\mu-3)^3}{(3+2\,\mu)^3}$

Table 3.3: Values of $g(\theta)$ for some θ, $p=2,\ s=3$.

θ	$g(\theta)$
$-\pi$	$\frac{(-2+\mu)^4}{(2+\mu)^4}$
$-\frac{3\pi}{4}$	$\frac{\left(-8+\mu\,\sqrt{2}+2\,\mu\right)^4}{\left(8+\mu\,\sqrt{2}+2\,\mu\right)^4}$
$-\frac{\pi}{2}$	$\frac{(-4+\mu)^4}{(4+\mu)^4}$
$-\frac{\pi}{4}$	$\frac{\left(8+\mu\,\sqrt{2}-2\,\mu\right)^4}{\left(-8+\mu\,\sqrt{2}-2\,\mu\right)^4}$
0	1
$\frac{\pi}{4}$	$\frac{\left(8+\mu\,\sqrt{2}-2\,\mu\right)^4}{\left(-8+\mu\,\sqrt{2}-2\,\mu\right)^4}$
$\frac{\pi}{2}$	$\frac{(-4+\mu)^4}{(4+\mu)^4}$
$\frac{3\pi}{4}$	$\frac{\left(-8+\mu\,\sqrt{2}+2\,\mu\right)^4}{\left(8+\mu\,\sqrt{2}+2\,\mu\right)^4}$
π	$\frac{(-2+\mu)^4}{(2+\mu)^4}$

Table 3.4: Values of $g(\theta)$ for some θ, $p=2,\ s=4$.

θ	$g(\theta)$
$-\pi$	$-\frac{\left(1+\sqrt{3}\right)\left(-3+4\,\mu\,\sqrt{3}-16\,\mu^2-3\,\sqrt{3}+12\,\mu\right)}{\left(3+4\,\mu+\sqrt{3}\right)^2}$
$-\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3+\mu\,\sqrt{3}\sqrt{2}+2\,\mu\,\sqrt{3}-6\,\mu^2-4\,\mu^2\sqrt{2}-3\,\sqrt{3}+3\,\mu\,\sqrt{2}+6\,\mu\right)}{\left(3+\mu\,\sqrt{2}+2\,\mu+\sqrt{3}\right)^2}$
$-\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3+2\,\mu\,\sqrt{3}-4\,\mu^2-3\,\sqrt{3}+6\,\mu\right)}{\left(3+2\,\mu+\sqrt{3}\right)^2}$
$-\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(3+\mu\,\sqrt{3}\sqrt{2}-2\,\mu\,\sqrt{3}+6\,\mu^2-4\,\mu^2\sqrt{2}+3\,\sqrt{3}+3\,\mu\,\sqrt{2}-6\,\mu\right)}{\left(3-\mu\,\sqrt{2}+2\,\mu+\sqrt{3}\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(3+\mu\,\sqrt{3}\sqrt{2}-2\,\mu\,\sqrt{3}+6\,\mu^2-4\,\mu^2\sqrt{2}+3\,\sqrt{3}+3\,\mu\,\sqrt{2}-6\,\mu\right)}{\left(3-\mu\,\sqrt{2}+2\,\mu+\sqrt{3}\right)^2}$
$\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3+2\,\mu\,\sqrt{3}-4\,\mu^2-3\,\sqrt{3}+6\,\mu\right)}{\left(3+2\,\mu+\sqrt{3}\right)^2}$
$\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3+\mu\,\sqrt{3}\sqrt{2}+2\,\mu\,\sqrt{3}-6\,\mu^2-4\,\mu^2\sqrt{2}-3\,\sqrt{3}+3\,\mu\,\sqrt{2}+6\,\mu\right)}{\left(3+\mu\,\sqrt{2}+2\,\mu+\sqrt{3}\right)^2}$
π	$-\frac{\left(1+\sqrt{3}\right)\left(-3+4\,\mu\,\sqrt{3}-16\,\mu^2-3\,\sqrt{3}+12\,\mu\right)}{\left(3+4\,\mu+\sqrt{3}\right)^2}$

Table 3.5: Values of $g(\theta)$ for some θ, $p = 3,\ s = 2$.

In Table 3.6, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$.
$\underline{p = 3:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed by MAPLE for $s = 4$: We do not reproduce the general expression for g here because it is too complicated for proper typesetting.
In Table 3.7, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$.

3.3.2 Fourth-order differences, Version 1 'Old ANTARES Code'

We perform the dissipativity analysis for a space discretisation resulting from

$$\begin{aligned} f_j &:= (L_h^{[1]}u)_j = \frac{1}{12\Delta x}\left(-u_{j+2} + 8u_{j+1} - 8u_{j-1} + u_{j-2}\right), \\ \hat{f}_{j+1/2} &= \frac{1}{16}(-f_{j-1} + 9f_j + 9f_{j+1} - f_{j+2}), \\ \hat{f}_{j-1/2} &= \frac{1}{16}(-f_{j-2} + 9f_{j-1} + 9f_j - f_{j+1}), \\ (L_h^{[2]}f)_j &= \frac{\hat{f}_{j+1/2} - \hat{f}_{j-1/2}}{\Delta x} \\ &= \frac{1}{16\Delta x}\left(-f_{j+2} + 10f_{j+1} - 10f_{j-1} + f_{j-2}\right), \\ u''(x_i) &\approx (L_h^{[2]}L_h^{[1]}u)_i, \end{aligned} \tag{3.40}$$

see also (3.17) and the discussion in Section 3.2.4. Thus, for $\mu = b\frac{\Delta t}{(\Delta x)^2}$ we analyze the relationship

$$g(\theta) = R\left(\frac{\mu}{192}\left(\mathrm{e}^{4\mathrm{i}\theta} - 18\mathrm{e}^{3\mathrm{i}\theta} + 80\mathrm{e}^{2\mathrm{i}\theta} + 18\mathrm{e}^{\mathrm{i}\theta} - 162 + 18\mathrm{e}^{-\mathrm{i}\theta} + 80\mathrm{e}^{-2\mathrm{i}\theta} - 18\mathrm{e}^{-3\mathrm{i}\theta} + \mathrm{e}^{-4\mathrm{i}\theta}\right)\right). \tag{3.41}$$

We obtained the following results:
$\underline{p = 2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 1$.
Here and in the following analysis of this scheme, we do not reproduce the general expression for g because it is too complicated for proper typesetting. In Table 3.8, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = g(\pm\pi) = 1$ and $|g(\theta)| < 1,\ \theta \notin \{-\pi, 0, \pi\}$.

θ	$g(\theta)$
$-\pi$	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(66-138\,\mu-108\,\mu\sqrt{2}+132\,\mu^2-68\,\mu^3+45\sqrt{2}+90\sqrt{2}\mu^2\right)}{\left(2+2\,\mu+\sqrt{2}\right)^3}$
$-\frac{3\pi}{4}$	$-\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-528+984\,\mu+708\,\mu\sqrt{2}-756\,\mu^2-534\sqrt{2}\mu^2+170\,\mu^3+119\,\mu^3\sqrt{2}-360\sqrt{2}\right)}{\left(4+2\,\mu+\mu\sqrt{2}+2\sqrt{2}\right)^3}$
$-\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(132-138\,\mu-108\,\mu\sqrt{2}+66\,\mu^2-17\,\mu^3+90\sqrt{2}+45\sqrt{2}\mu^2\right)}{\left(2+\mu+\sqrt{2}\right)^3}$
$-\frac{\pi}{4}$	$-\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(528-156\,\mu\sqrt{2}-120\,\mu+36\,\mu^2+6\sqrt{2}\mu^2+119\,\mu^3\sqrt{2}-170\,\mu^3+360\sqrt{2}\right)}{\left(-4+\mu\sqrt{2}-2\,\mu-2\sqrt{2}\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(528-156\,\mu\sqrt{2}-120\,\mu+36\,\mu^2+6\sqrt{2}\mu^2+119\,\mu^3\sqrt{2}-170\,\mu^3+360\sqrt{2}\right)}{\left(-4+\mu\sqrt{2}-2\,\mu-2\sqrt{2}\right)^3}$
$\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(132-138\,\mu-108\,\mu\sqrt{2}+66\,\mu^2-17\,\mu^3+90\sqrt{2}+45\sqrt{2}\mu^2\right)}{\left(2+\mu+\sqrt{2}\right)^3}$
$\frac{3\pi}{4}$	$-\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-528+984\,\mu+708\,\mu\sqrt{2}-756\,\mu^2-534\sqrt{2}\mu^2+170\,\mu^3+119\,\mu^3\sqrt{2}-360\sqrt{2}\right)}{\left(4+2\,\mu+\mu\sqrt{2}+2\sqrt{2}\right)^3}$
π	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(66-138\,\mu-108\,\mu\sqrt{2}+132\,\mu^2-68\,\mu^3+45\sqrt{2}+90\sqrt{2}\mu^2\right)}{\left(2+2\,\mu+\sqrt{2}\right)^3}$

Table 3.6: Values of $g(\theta)$ for some θ, $p=3,\ s=3$.

$\underline{p=2}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=2$.
In Table 3.9, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)|\le 1$ holds for all $\mu>0$, with $g(0)=g(\pm\pi)=1$ and $|g(\theta)|<1,\ \theta\notin\{-\pi,0,\pi\}$.
$\underline{p=2}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=3$.
In Table 3.10, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)|\le 1$ holds for all $\mu>0$, with $g(0)=g(\pm\pi)=1$ and $|g(\theta)|<1,\ \theta\notin\{-\pi,0,\pi\}$.
$\underline{p=2}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$.
In Table 3.11, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)|\le 1$ holds for all $\mu>0$, with $g(0)=g(\pm\pi)=1$ and $|g(\theta)|<1,\ \theta\notin\{-\pi,0,\pi\}$.
$\underline{p=3}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=2$.
In Table 3.12, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)|<1$. The condition is violated however for $\theta=\pm\pi$.
$\underline{p=3}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=3$.
In Table 3.13, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)|<1$. The condition is violated however for $\theta=\pm\pi$.
$\underline{p=3}$: With $\mu=b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$.
We evaluated $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$, and in Table 3.14 we observed the following behavior: The condition $|g(\theta)|<1$ is violated for $\theta=\pm\pi$, and otherwise it appears that there are certain restrictions on μ if we require $|g(\theta)|<1$. We display the results only with coefficients rounded to the third digit, because the formulae are too complicated for typesetting.

θ	$g(\theta)$
$-\pi$	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075+33300\,\mu+8520\,\mu\sqrt{15}-3900\sqrt{15}-29880\,\mu^2+11040\,\mu^3-3392\,\mu^4-7920\,\mu^2\sqrt{15}+3008\,\mu^3\sqrt{15}\right)}{\left(5+4\,\mu+\sqrt{15}\right)^4}$
$-\frac{3\pi}{4}$	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075-3900\sqrt{15}-11205\,\mu^2+3450\,\mu^3-901\,\mu^4+4260\,\mu\sqrt{15}+16650\,\mu+2130\,\mu\sqrt{15}\sqrt{2}-1980\,\mu^2\sqrt{15}\sqrt{2}+658\,\mu^3\sqrt{15}\sqrt{2}+8325\,\mu\sqrt{2}-7470\,\mu^2\sqrt{2}+2415\,\mu^3\sqrt{2}-636\,\mu^4\sqrt{2}+940\,\mu^3\sqrt{15}-2970\,\mu^2\sqrt{15}\right)}{\left(5+\mu\sqrt{2}+2\,\mu+\sqrt{15}\right)^4}$
$-\frac{\pi}{2}$	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075+16650\,\mu+4260\,\mu\sqrt{15}-3900\sqrt{15}-7470\,\mu^2+1380\,\mu^3-212\,\mu^4-1980\,\mu^2\sqrt{15}+376\,\mu^3\sqrt{15}\right)}{\left(5+2\,\mu+\sqrt{15}\right)^4}$
$-\frac{\pi}{4}$	$\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(15075+3900\sqrt{15}+11205\,\mu^2-3450\,\mu^3+901\,\mu^4-4260\,\mu\sqrt{15}-16650\,\mu+2130\,\mu\sqrt{15}\sqrt{2}-1980\,\mu^2\sqrt{15}\sqrt{2}+658\,\mu^3\sqrt{15}\sqrt{2}+8325\,\mu\sqrt{2}-7470\,\mu^2\sqrt{2}+2415\,\mu^3\sqrt{2}-636\,\mu^4\sqrt{2}-940\,\mu^3\sqrt{15}+2970\,\mu^2\sqrt{15}\right)}{\left(5-\mu\sqrt{2}+2\,\mu+\sqrt{15}\right)^4}$
0	1
$\frac{\pi}{4}$	$\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(15075+3900\sqrt{15}+11205\,\mu^2-3450\,\mu^3+901\,\mu^4-4260\,\mu\sqrt{15}-16650\,\mu+2130\,\mu\sqrt{15}\sqrt{2}-1980\,\mu^2\sqrt{15}\sqrt{2}+658\,\mu^3\sqrt{15}\sqrt{2}+8325\,\mu\sqrt{2}-7470\,\mu^2\sqrt{2}+2415\,\mu^3\sqrt{2}-636\,\mu^4\sqrt{2}-940\,\mu^3\sqrt{15}+2970\,\mu^2\sqrt{15}\right)}{\left(5-\mu\sqrt{2}+2\,\mu+\sqrt{15}\right)^4}$
$\frac{\pi}{2}$	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075+16650\,\mu+4260\,\mu\sqrt{15}-3900\sqrt{15}-7470\,\mu^2+1380\,\mu^3-212\,\mu^4-1980\,\mu^2\sqrt{15}+376\,\mu^3\sqrt{15}\right)}{\left(5+2\,\mu+\sqrt{15}\right)^4}$
$\frac{3\pi}{4}$	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075-3900\sqrt{15}-11205\,\mu^2+3450\,\mu^3-901\,\mu^4+4260\,\mu\sqrt{15}+16650\,\mu+2130\,\mu\sqrt{15}\sqrt{2}-1980\,\mu^2\sqrt{15}\sqrt{2}+658\,\mu^3\sqrt{15}\sqrt{2}+8325\,\mu\sqrt{2}-7470\,\mu^2\sqrt{2}+2415\,\mu^3\sqrt{2}-636\,\mu^4\sqrt{2}+940\,\mu^3\sqrt{15}-2970\,\mu^2\sqrt{15}\right)}{\left(5+\mu\sqrt{2}+2\,\mu+\sqrt{15}\right)^4}$
π	$-\frac{4}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-15075+33300\,\mu+8520\,\mu\sqrt{15}-3900\sqrt{15}-29880\,\mu^2+11040\,\mu^3-3392\,\mu^4-7920\,\mu^2\sqrt{15}+3008\,\mu^3\sqrt{15}\right)}{\left(5+4\,\mu+\sqrt{15}\right)^4}$

Table 3.7: Values of $g(\theta)$ for some θ, $p = 3,\ s = 4$.

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{-96+9\,\mu\,\sqrt{2}+41\,\mu}{96+9\,\mu\,\sqrt{2}+41\,\mu}$
$-\frac{\pi}{2}$	$-\frac{-6+5\,\mu}{6+5\,\mu}$
$-\frac{\pi}{4}$	$-\frac{96+9\,\mu\,\sqrt{2}-41\,\mu}{-96+9\,\mu\,\sqrt{2}-41\,\mu}$
0	1
$\frac{\pi}{4}$	$-\frac{96+9\,\mu\,\sqrt{2}-41\,\mu}{-96+9\,\mu\,\sqrt{2}-41\,\mu}$
$\frac{\pi}{2}$	$-\frac{-6+5\,\mu}{6+5\,\mu}$
$\frac{3\pi}{4}$	$-\frac{-96+9\,\mu\,\sqrt{2}+41\,\mu}{96+9\,\mu\,\sqrt{2}+41\,\mu}$
π	1

Table 3.8: Values of $g(\theta)$ for some θ, $p=2,\ s=1$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$\frac{\left(9\,\mu\,\sqrt{2}+41\,\mu-192\right)^2}{\left(192+9\,\mu\,\sqrt{2}+41\,\mu\right)^2}$
$-\frac{\pi}{2}$	$\frac{(5\,\mu-12)^2}{(12+5\,\mu)^2}$
$-\frac{\pi}{4}$	$\frac{\left(9\,\mu\,\sqrt{2}-41\,\mu+192\right)^2}{\left(-192+9\,\mu\,\sqrt{2}-41\,\mu\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(9\,\mu\,\sqrt{2}-41\,\mu+192\right)^2}{\left(-192+9\,\mu\,\sqrt{2}-41\,\mu\right)^2}$
$\frac{\pi}{2}$	$\frac{(5\,\mu-12)^2}{(12+5\,\mu)^2}$
$\frac{3\pi}{4}$	$\frac{\left(9\,\mu\,\sqrt{2}+41\,\mu-192\right)^2}{\left(192+9\,\mu\,\sqrt{2}+41\,\mu\right)^2}$
π	1

Table 3.9: Values of $g(\theta)$ for some θ, $p=2,\ s=2$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{\left(9\,\mu\,\sqrt{2}+41\,\mu-288\right)^3}{\left(288+9\,\mu\,\sqrt{2}+41\,\mu\right)^3}$
$-\frac{\pi}{2}$	$-\frac{(5\,\mu-18)^3}{(18+5\,\mu)^3}$
$-\frac{\pi}{4}$	$-\frac{\left(9\,\mu\,\sqrt{2}-41\,\mu+288\right)^3}{\left(-288+9\,\mu\,\sqrt{2}-41\,\mu\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{\left(9\,\mu\,\sqrt{2}-41\,\mu+288\right)^3}{\left(-288+9\,\mu\,\sqrt{2}-41\,\mu\right)^3}$
$\frac{\pi}{2}$	$-\frac{(5\,\mu-18)^3}{(18+5\,\mu)^3}$
$\frac{3\pi}{4}$	$-\frac{\left(9\,\mu\,\sqrt{2}+41\,\mu-288\right)^3}{\left(288+9\,\mu\,\sqrt{2}+41\,\mu\right)^3}$
π	1

Table 3.10: Values of $g(\theta)$ for some θ, $p=2,\ s=3$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$\frac{\left(-384+9\,\mu\,\sqrt{2}+41\,\mu\right)^4}{\left(384+9\,\mu\,\sqrt{2}+41\,\mu\right)^4}$
$-\frac{\pi}{2}$	$\frac{\left(-24+5\,\mu\right)^4}{\left(24+5\,\mu\right)^4}$
$-\frac{\pi}{4}$	$\frac{\left(384+9\,\mu\,\sqrt{2}-41\,\mu\right)^4}{\left(-384+9\,\mu\,\sqrt{2}-41\,\mu\right)^4}$
0	1
$\frac{\pi}{4}$	$\frac{\left(384+9\,\mu\,\sqrt{2}-41\,\mu\right)^4}{\left(-384+9\,\mu\,\sqrt{2}-41\,\mu\right)^4}$
$\frac{\pi}{2}$	$\frac{\left(-24+5\,\mu\right)^4}{\left(24+5\,\mu\right)^4}$
$\frac{3\pi}{4}$	$\frac{\left(-384+9\,\mu\,\sqrt{2}+41\,\mu\right)^4}{\left(384+9\,\mu\,\sqrt{2}+41\,\mu\right)^4}$
π	1

Table 3.11: Values of $g(\theta)$ for some θ, $p=2,\ s=4$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-6912+432\,\mu\,\sqrt{3}\sqrt{2}+1968\,\mu\,\sqrt{3}-1843\,\mu^2-738\,\mu^2\sqrt{2}-6912\,\sqrt{3}+1296\,\mu\,\sqrt{2}+5904\,\mu\right)}{\left(144+9\,\mu\,\sqrt{2}+41\,\mu+48\,\sqrt{3}\right)^2}$
$-\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-27+15\,\mu\,\sqrt{3}-25\,\mu^2-27\,\sqrt{3}+45\,\mu\right)}{\left(9+5\,\mu+3\,\sqrt{3}\right)^2}$
$-\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(6912+432\,\mu\,\sqrt{3}\sqrt{2}-1968\,\mu\,\sqrt{3}+1843\,\mu^2-738\,\mu^2\sqrt{2}+6912\,\sqrt{3}+1296\,\mu\,\sqrt{2}-5904\,\mu\right)}{\left(144-9\,\mu\,\sqrt{2}+41\,\mu+48\,\sqrt{3}\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(6912+432\,\mu\,\sqrt{3}\sqrt{2}-1968\,\mu\,\sqrt{3}+1843\,\mu^2-738\,\mu^2\sqrt{2}+6912\,\sqrt{3}+1296\,\mu\,\sqrt{2}-5904\,\mu\right)}{\left(144-9\,\mu\,\sqrt{2}+41\,\mu+48\,\sqrt{3}\right)^2}$
$\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-27+15\,\mu\,\sqrt{3}-25\,\mu^2-27\,\sqrt{3}+45\,\mu\right)}{\left(9+5\,\mu+3\,\sqrt{3}\right)^2}$
$\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-6912+432\,\mu\,\sqrt{3}\sqrt{2}+1968\,\mu\,\sqrt{3}-1843\,\mu^2-738\,\mu^2\sqrt{2}-6912\,\sqrt{3}+1296\,\mu\,\sqrt{2}+5904\,\mu\right)}{\left(144+9\,\mu\,\sqrt{2}+41\,\mu+48\,\sqrt{3}\right)^2}$
π	1

Table 3.12: Values of $g(\theta)$ for some θ, $p=3,\ s=2$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-116785152+52254720\,\mu\sqrt{2}+70060032\,\mu-18053568\,\mu^2-12637728\sqrt{2}\mu^2+796365\,\mu^3\sqrt{2}+1510399\,\mu^3-79626240\sqrt{2}\right)}{\left(192+9\,\mu\sqrt{2}+41\,\mu+96\sqrt{2}\right)^3}$
$-\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(28512-24840\,\mu-19440\,\mu\sqrt{2}+9900\,\mu^2-2125\,\mu^3+19440\sqrt{2}+6750\sqrt{2}\mu^2\right)}{\left(12+5\,\mu+6\sqrt{2}\right)^3}$
$-\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(116785152-29362176\,\mu\sqrt{2}-34228224\,\mu+5300928\,\mu^2+3285792\sqrt{2}\mu^2+796365\,\mu^3\sqrt{2}-1510399\,\mu^3+79626240\sqrt{2}\right)}{\left(-192+9\,\mu\sqrt{2}-41\,\mu-96\sqrt{2}\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(116785152-29362176\,\mu\sqrt{2}-34228224\,\mu+5300928\,\mu^2+3285792\sqrt{2}\mu^2+796365\,\mu^3\sqrt{2}-1510399\,\mu^3+79626240\sqrt{2}\right)}{\left(-192+9\,\mu\sqrt{2}-41\,\mu-96\sqrt{2}\right)^3}$
$\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(28512-24840\,\mu-19440\,\mu\sqrt{2}+9900\,\mu^2-2125\,\mu^3+19440\sqrt{2}+6750\sqrt{2}\mu^2\right)}{\left(12+5\,\mu+6\sqrt{2}\right)^3}$
$\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-116785152+52254720\,\mu\sqrt{2}+70060032\,\mu-18053568\,\mu^2-12637728\sqrt{2}\mu^2+796365\,\mu^3\sqrt{2}+1510399\,\mu^3-79626240\sqrt{2}\right)}{\left(192+9\,\mu\sqrt{2}+41\,\mu+96\sqrt{2}\right)^3}$
π	1

Table 3.13: Values of $g(\theta)$ for some θ, $p=3,\ s=3$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-0.052\,\frac{10000000000.0\,\mu^3-1.00\times10^{11}\,\mu^2-6.4\times10^{11}+4.00\times10^{11}\,\mu-440000000.0\,\mu^4}{(430.0+54.0\,\mu)^4}$
$-\frac{\pi}{2}$	$-\frac{25}{477}\,\frac{\left(9+4\sqrt{15}\right)\left(-195372+179820\,\mu-50544\sqrt{15}+46008\,\mu\sqrt{15}-67230\,\mu^2+10350\,\mu^3-1325\,\mu^4+2820\,\mu^3\sqrt{15}-17820\,\mu^2\sqrt{15}\right)}{\left(15+5\,\mu+3\sqrt{15}\right)^4}$
$-\frac{\pi}{4}$	$0.052\,\frac{6.4\times10^{11}+28000000000.0\,\mu^2-2.00\times10^{11}\,\mu-1500000000.0\,\mu^3+40000000.0\,\mu^4}{(430.0+28.0\,\mu)^4}$
0	1
$\frac{\pi}{4}$	$0.052\,\frac{6.4\times10^{11}-1500000000.0\,\mu^3-2.00\times10^{11}\,\mu+28000000000.0\,\mu^2+40000000.0\,\mu^4}{(-430.0-28.0\,\mu)^4}$
$\frac{\pi}{2}$	$-1.3\,\frac{-400000.0+360000.0\,\mu-140000.0\,\mu^2+21000.0\,\mu^3-1300.0\,\mu^4}{(27.0+5.0\,\mu)^4}$
$\frac{3\pi}{4}$	$-0.052\,\frac{-6.4\times10^{11}+4.00\times10^{11}\,\mu+10000000000.0\,\mu^3-440000000.0\,\mu^4-1.00\times10^{11}\,\mu^2}{(430.0+54.0\,\mu)^4}$
π	1

Table 3.14: Values of $g(\theta)$ for some θ, $p=3,\ s=4$, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{-72+8\,\mu\,\sqrt{2}+33\,\mu}{72+8\,\mu\,\sqrt{2}+33\,\mu}$
$-\frac{\pi}{2}$	$-\frac{-9+8\,\mu}{9+8\,\mu}$
$-\frac{\pi}{4}$	$-\frac{72+8\,\mu\,\sqrt{2}-33\,\mu}{-72+8\,\mu\,\sqrt{2}-33\,\mu}$
0	1
$\frac{\pi}{4}$	$-\frac{72+8\,\mu\,\sqrt{2}-33\,\mu}{-72+8\,\mu\,\sqrt{2}-33\,\mu}$
$\frac{\pi}{2}$	$-\frac{-9+8\,\mu}{9+8\,\mu}$
$\frac{3\pi}{4}$	$-\frac{-72+8\,\mu\,\sqrt{2}+33\,\mu}{72+8\,\mu\,\sqrt{2}+33\,\mu}$
π	1

Table 3.15: Values of $g(\theta)$ for some θ, $p = 2,\ s = 1$, fourth order (old ANTARES 'version 2').

3.3.3 Fourth-order differences, Version 2 'Old ANTARES Code'

We perform the dissipativity analysis for a space discretisation resulting from

$$\begin{aligned} f_j &:= (L_h^{[1]}u)_j = \frac{1}{12\Delta x}\left(-u_{j+2}+8u_{j+1}-8u_{j-1}+u_{j-2}\right), \\ \hat{f}_{j+1/2} &= \frac{1}{12}(-f_{j-1}+7f_j+7f_{j+1}-f_{j+2}), \\ \hat{f}_{j-1/2} &= \frac{1}{12}(-f_{j-2}+7f_{j-1}+7f_j-f_{j+1}), \\ (L_h^{[2]}f)_j &= \frac{\hat{f}_{j+1/2}-\hat{f}_{j-1/2}}{\Delta x} = (L_h^{[1]}f)_j, \\ u''(x_i) &\approx ((L_h^{[1]})^2u)_i, \end{aligned} \tag{3.42}$$

see also (3.18) and the discussion in Section 3.2.4.
Thus, for $\mu = b\frac{\Delta t}{(\Delta x)^2}$ we analyze the relationship

$$g(\theta) = R\left(\frac{\mu}{144}\left(\mathrm{e}^{4\mathrm{i}\theta}-16\mathrm{e}^{3\mathrm{i}\theta}+64\mathrm{e}^{2\mathrm{i}\theta}+16\mathrm{e}^{\mathrm{i}\theta}-130+16\mathrm{e}^{-\mathrm{i}\theta}+64\mathrm{e}^{-2\mathrm{i}\theta}-16\mathrm{e}^{-3\mathrm{i}\theta}+\mathrm{e}^{-4\mathrm{i}\theta}\right)\right). \tag{3.43}$$

We obtained the following results:
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 1$. Here and in the following analysis of this scheme, we do not reproduce the general expression for g because it is too complicated for proper typesetting.
In Table 3.15, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = g(\pm\pi) = 1$ and $|g(\theta)| < 1,\ \theta \notin \{-\pi, 0, \pi\}$.
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 2$.
In Table 3.16, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = g(\pm\pi) = 1$ and $|g(\theta)| < 1,\ \theta \notin \{-\pi, 0, \pi\}$.
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 3$.
In Table 3.17, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = g(\pm\pi) = 1$ and $|g(\theta)| < 1,\ \theta \notin \{-\pi, 0, \pi\}$.
For the values of θ we computed, the condition $|g(\theta)| < 1$ is violated for $\mu \in \{-\pi, 0, \pi\}$, but holds otherwise.
$\underline{p=2:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 4$.
In Table 3.18, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = g(\pm\pi) = 1$ and $|g(\theta)| < 1,\ \theta \notin \{-\pi, 0, \pi\}$.
$\underline{p=3:}$ With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s = 2$.

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$\frac{\left(8\,\mu\,\sqrt{2}+33\,\mu-144\right)^2}{\left(144+8\,\mu\,\sqrt{2}+33\,\mu\right)^2}$
$-\frac{\pi}{2}$	$\frac{(4\,\mu-9)^2}{(9+4\,\mu)^2}$
$-\frac{\pi}{4}$	$\frac{\left(8\,\mu\,\sqrt{2}-33\,\mu+144\right)^2}{\left(-144+8\,\mu\,\sqrt{2}-33\,\mu\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(8\,\mu\,\sqrt{2}-33\,\mu+144\right)^2}{\left(-144+8\,\mu\,\sqrt{2}-33\,\mu\right)^2}$
$\frac{\pi}{2}$	$\frac{(4\,\mu-9)^2}{(9+4\,\mu)^2}$
$\frac{3\pi}{4}$	$\frac{\left(8\,\mu\,\sqrt{2}+33\,\mu-144\right)^2}{\left(144+8\,\mu\,\sqrt{2}+33\,\mu\right)^2}$
π	1

Table 3.16: Values of $g(\theta)$ for some θ, $p = 2,\ s = 2$, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{\left(8\,\mu\,\sqrt{2}+33\,\mu-216\right)^3}{\left(216+8\,\mu\,\sqrt{2}+33\,\mu\right)^3}$
$-\frac{\pi}{2}$	$-\frac{(8\,\mu-27)^3}{(27+8\,\mu)^3}$
$-\frac{\pi}{4}$	$-\frac{\left(8\,\mu\,\sqrt{2}-33\,\mu+216\right)^3}{\left(-216+8\,\mu\,\sqrt{2}-33\,\mu\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{\left(8\,\mu\,\sqrt{2}-33\,\mu+216\right)^3}{\left(-216+8\,\mu\,\sqrt{2}-33\,\mu\right)^3}$
$\frac{\pi}{2}$	$-\frac{(8\,\mu-27)^3}{(27+8\,\mu)^3}$
$\frac{3\pi}{4}$	$-\frac{\left(8\,\mu\,\sqrt{2}+33\,\mu-216\right)^3}{\left(216+8\,\mu\,\sqrt{2}+33\,\mu\right)^3}$
π	1

Table 3.17: Values of $g(\theta)$ for some θ, $p = 2,\ s = 3$, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$\frac{\left(-288+8\,\mu\,\sqrt{2}+33\,\mu\right)^4}{\left(288+8\,\mu\,\sqrt{2}+33\,\mu\right)^4}$
$-\frac{\pi}{2}$	$\frac{(-9+2\,\mu)^4}{(9+2\,\mu)^4}$
$-\frac{\pi}{4}$	$\frac{\left(288+8\,\mu\,\sqrt{2}-33\,\mu\right)^4}{\left(-288+8\,\mu\,\sqrt{2}-33\,\mu\right)^4}$
0	1
$\frac{\pi}{4}$	$\frac{\left(288+8\,\mu\,\sqrt{2}-33\,\mu\right)^4}{\left(-288+8\,\mu\,\sqrt{2}-33\,\mu\right)^4}$
$\frac{\pi}{2}$	$\frac{(-9+2\,\mu)^4}{(9+2\,\mu)^4}$
$\frac{3\pi}{4}$	$\frac{\left(-288+8\,\mu\,\sqrt{2}+33\,\mu\right)^4}{\left(288+8\,\mu\,\sqrt{2}+33\,\mu\right)^4}$
π	1

Table 3.18: Values of $g(\theta)$ for some θ, $p=2,\ s=4$, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3888+288\,\mu\,\sqrt{3}\sqrt{2}+1188\,\mu\,\sqrt{3}-1217\,\mu^2-528\,\mu^2\sqrt{2}-3888\,\sqrt{3}+864\,\mu\,\sqrt{2}+3564\,\mu\right)}{\left(108+8\,\mu\,\sqrt{2}+33\,\mu+36\,\sqrt{3}\right)^2}$
$-\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-243+144\,\mu\,\sqrt{3}-256\,\mu^2-243\,\sqrt{3}+432\,\mu\right)}{\left(27+16\,\mu+9\,\sqrt{3}\right)^2}$
$-\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(3888+288\,\mu\,\sqrt{3}\sqrt{2}-1188\,\mu\,\sqrt{3}+1217\,\mu^2-528\,\mu^2\sqrt{2}+3888\,\sqrt{3}+864\,\mu\,\sqrt{2}-3564\,\mu\right)}{\left(108-8\,\mu\,\sqrt{2}+33\,\mu+36\,\sqrt{3}\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(3888+288\,\mu\,\sqrt{3}\sqrt{2}-1188\,\mu\,\sqrt{3}+1217\,\mu^2-528\,\mu^2\sqrt{2}+3888\,\sqrt{3}+864\,\mu\,\sqrt{2}-3564\,\mu\right)}{\left(108-8\,\mu\,\sqrt{2}+33\,\mu+36\,\sqrt{3}\right)^2}$
$\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-243+144\,\mu\,\sqrt{3}-256\,\mu^2-243\,\sqrt{3}+432\,\mu\right)}{\left(27+16\,\mu+9\,\sqrt{3}\right)^2}$
$\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-3888+288\,\mu\,\sqrt{3}\sqrt{2}+1188\,\mu\,\sqrt{3}-1217\,\mu^2-528\,\mu^2\sqrt{2}-3888\,\sqrt{3}+864\,\mu\,\sqrt{2}+3564\,\mu\right)}{\left(108+8\,\mu\,\sqrt{2}+33\,\mu+36\,\sqrt{3}\right)^2}$
π	1

Table 3.19: Values of $g(\theta)$ for some θ, $p=3,\ s=2$, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-49268736+24198912\,\mu\sqrt{2}+32565888\,\mu-9204624\,\mu^2-6452136\sqrt{2}\mu^2+461720\,\mu^3\sqrt{2}+826353\,\mu^3-33592320\sqrt{2}\right)}{\left(144+8\,\mu\sqrt{2}+33\,\mu+72\sqrt{2}\right)^3}$
$-\frac{\pi}{2}$	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(48114-44712\,\mu-34992\,\mu\sqrt{2}+19008\,\mu^2-4352\,\mu^3+32805\sqrt{2}+12960\sqrt{2}\mu^2\right)}{\left(18+8\,\mu+9\sqrt{2}\right)^3}$
$-\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(49268736-12752640\,\mu\sqrt{2}-14649984\,\mu+2361744\,\mu^2+1434024\sqrt{2}\mu^2+461720\,\mu^3\sqrt{2}-826353\,\mu^3+33592320\sqrt{2}\right)}{\left(-144+8\,\mu\sqrt{2}-33\,\mu-72\sqrt{2}\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(49268736-12752640\,\mu\sqrt{2}-14649984\,\mu+2361744\,\mu^2+1434024\sqrt{2}\mu^2+461720\,\mu^3\sqrt{2}-826353\,\mu^3+33592320\sqrt{2}\right)}{\left(-144+8\,\mu\sqrt{2}-33\,\mu-72\sqrt{2}\right)^3}$
$\frac{\pi}{2}$	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(48114-44712\,\mu-34992\,\mu\sqrt{2}+19008\,\mu^2-4352\,\mu^3+32805\sqrt{2}+12960\sqrt{2}\mu^2\right)}{\left(18+8\,\mu+9\sqrt{2}\right)^3}$
$\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-49268736+24198912\,\mu\sqrt{2}+32565888\,\mu-9204624\,\mu^2-6452136\sqrt{2}\mu^2+461720\,\mu^3\sqrt{2}+826353\,\mu^3-33592320\sqrt{2}\right)}{\left(144+8\,\mu\sqrt{2}+33\,\mu+72\sqrt{2}\right)^3}$
π	1

Table 3.20: Values of $g(\theta)$ for some θ, $p=3,\ s=3$, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$-0.052\,\frac{-2.00\times10^{11}+1.4\times10^{11}\,\mu-39000000000.0\,\mu^2+4400000000.0\,\mu^3-200000000.0\,\mu^4}{(320.0+44.0\,\mu)^4}$
$-\frac{\pi}{2}$	$-0.21\,\frac{-200000000.0+200000000.0\,\mu-78000000.0\,\mu^2+13000000.0\,\mu^3-870000.0\,\mu^4}{(80.0+16.0\,\mu)^4}$
$-\frac{\pi}{4}$	$0.052\,\frac{2.00\times10^{11}-67000000000.0\,\mu+9000000000.0\,\mu^2-600000000.0\,\mu^3+15000000.0\,\mu^4}{(-320.0-22.0\,\mu)^4}$
0	1
$\frac{\pi}{4}$	$0.052\,\frac{2.00\times10^{11}-67000000000.0\,\mu+9000000000.0\,\mu^2-600000000.0\,\mu^3+15000000.0\,\mu^4}{(-320.0-22.0\,\mu)^4}$
$\frac{\pi}{2}$	$-0.21\,\frac{-200000000.0+200000000.0\,\mu-78000000.0\,\mu^2+13000000.0\,\mu^3-870000.0\,\mu^4}{(80.0+16.0\,\mu)^4}$
$\frac{3\pi}{4}$	$-0.052\,\frac{-2.00\times10^{11}+1.4\times10^{11}\,\mu-39000000000.0\,\mu^2+4400000000.0\,\mu^3-200000000.0\,\mu^4}{(320.0+44.0\,\mu)^4}$
π	1

Table 3.21: Values of $g(\theta)$ for some θ, $p=3,\ s=4$, fourth order (old ANTARES 'version 2').

In Table 3.19, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$. $g(\pm\pi) = 1$ holds irrespectively of μ.
$\underline{p=3}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=3$.
In Table 3.20, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$. $g(\pm\pi) = 1$ holds irrespectively of μ.
$\underline{p=3}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$.
In Table 3.21, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$, where the results are rounded to two significant digits: It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$. $g(0) = g(\pm\pi) = 1$ holds irrespectively of μ.

θ	$g(\theta)$
$-\pi$	$-\frac{-3+8\,\mu}{3+8\,\mu}$
$-\frac{3\pi}{4}$	$-\frac{-12+15\,\mu+8\,\mu\,\sqrt{2}}{12+15\,\mu+8\,\mu\,\sqrt{2}}$
$-\frac{\pi}{2}$	$-\frac{-6+7\,\mu}{6+7\,\mu}$
$-\frac{\pi}{4}$	$-\frac{12-15\,\mu+8\,\mu\,\sqrt{2}}{-12-15\,\mu+8\,\mu\,\sqrt{2}}$
0	1
$\frac{\pi}{4}$	$-\frac{12-15\,\mu+8\,\mu\,\sqrt{2}}{-12-15\,\mu+8\,\mu\,\sqrt{2}}$
$\frac{\pi}{2}$	$-\frac{-6+7\,\mu}{6+7\,\mu}$
$\frac{3\pi}{4}$	$-\frac{-12+15\,\mu+8\,\mu\,\sqrt{2}}{12+15\,\mu+8\,\mu\,\sqrt{2}}$
π	$-\frac{-3+8\,\mu}{3+8\,\mu}$

Table 3.22: Values of $g(\theta)$ for some θ, $p=2,\ s=1$, fourth order (new ANTARES).

3.3.4 Fourth-order differences, Version 'New ANTARES Code'

We perform the dissipativity analysis for the space discretisation (3.12) derived in Section 3.2.6,

$$\begin{aligned} f_j &:= (L_h^{[1]}u)_j = \frac{1}{12\Delta x}\left(u_{j-1} - 15u_j + 15u_{j+1} - u_{j+2}\right), \\ (L_h^{[2]}f)_j &= \frac{1}{\Delta x}\left(f_j - f_{j-1}\right), \\ u''(x_i) &\approx (L_h^{[2]}L_h^{[1]}u)_i. \end{aligned} \tag{3.44}$$

Thus, for $\mu = b\frac{\Delta t}{(\Delta x)^2}$ we analyze the relationship

$$g(\theta) = R\left(\frac{\mu}{12}\left(-\mathrm{e}^{-2\mathrm{i}\theta} + 16\mathrm{e}^{-\mathrm{i}\theta} - 30 + 16\mathrm{e}^{\mathrm{i}\theta} - \mathrm{e}^{2\mathrm{i}\theta}\right)\right). \tag{3.45}$$

We obtained the following results:
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=1$:

$$g(\theta) = -\frac{\mu\,(\cos(\theta))^2 + 7\,\mu - 8\,\mu\,\cos(\theta) - 6}{\mu\,(\cos(\theta))^2 + 7\,\mu - 8\,\mu\,\cos(\theta) + 6}. \tag{3.46}$$

In Table 3.22, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = 1$ and $|g(\theta)| < 1,\ \theta \neq 0$.
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=2$. Here and in the following analysis of this scheme, we do not reproduce the general expression for g because it is too complicated for proper typesetting.
In Table 3.23, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = 1$ and $|g(\theta)| < 1,\ \theta \neq 0$.
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=3$.
In Table 3.24, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = 1$ and $|g(\theta)| < 1,\ \theta \neq 0$.
$\underline{p=2}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$.
In Table 3.25, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$: For the values of θ we computed, $|g(\theta)| \le 1$ holds for all $\mu > 0$, with $g(0) = 1$ and $|g(\theta)| < 1,\ \theta \neq 0$.
$\underline{p=3}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=2$.
In Table 3.26, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$.
$\underline{p=3}$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=3$.

θ	$g(\theta)$
$-\pi$	$\frac{(4\,\mu-3)^2}{(3+4\,\mu)^2}$
$-\frac{3\pi}{4}$	$\frac{\left(15\,\mu+8\,\mu\,\sqrt{2}-24\right)^2}{\left(24+15\,\mu+8\,\mu\,\sqrt{2}\right)^2}$
$-\frac{\pi}{2}$	$\frac{(7\,\mu-12)^2}{(12+7\,\mu)^2}$
$-\frac{\pi}{4}$	$\frac{\left(-15\,\mu+8\,\mu\,\sqrt{2}+24\right)^2}{\left(-24-15\,\mu+8\,\mu\,\sqrt{2}\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(-15\,\mu+8\,\mu\,\sqrt{2}+24\right)^2}{\left(-24-15\,\mu+8\,\mu\,\sqrt{2}\right)^2}$
$\frac{\pi}{2}$	$\frac{(7\,\mu-12)^2}{(12+7\,\mu)^2}$
$\frac{3\pi}{4}$	$\frac{\left(15\,\mu+8\,\mu\,\sqrt{2}-24\right)^2}{\left(24+15\,\mu+8\,\mu\,\sqrt{2}\right)^2}$
π	$\frac{(4\,\mu-3)^2}{(3+4\,\mu)^2}$

Table 3.23: Values of $g(\theta)$ for some θ, $p=2,\ s=2$, fourth order (new ANTARES).

θ	$g(\theta)$
$-\pi$	$-\frac{(8\,\mu-9)^3}{(9+8\,\mu)^3}$
$-\frac{3\pi}{4}$	$-\frac{\left(15\,\mu+8\,\mu\,\sqrt{2}-36\right)^3}{\left(36+15\,\mu+8\,\mu\,\sqrt{2}\right)^3}$
$-\frac{\pi}{2}$	$-\frac{(7\,\mu-18)^3}{(18+7\,\mu)^3}$
$-\frac{\pi}{4}$	$-\frac{\left(-15\,\mu+8\,\mu\,\sqrt{2}+36\right)^3}{\left(-36-15\,\mu+8\,\mu\,\sqrt{2}\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{\left(-15\,\mu+8\,\mu\,\sqrt{2}+36\right)^3}{\left(-36-15\,\mu+8\,\mu\,\sqrt{2}\right)^3}$
$\frac{\pi}{2}$	$-\frac{(7\,\mu-18)^3}{(18+7\,\mu)^3}$
$\frac{3\pi}{4}$	$-\frac{\left(15\,\mu+8\,\mu\,\sqrt{2}-36\right)^3}{\left(36+15\,\mu+8\,\mu\,\sqrt{2}\right)^3}$
π	$-\frac{(8\,\mu-9)^3}{(9+8\,\mu)^3}$

Table 3.24: Values of $g(\theta)$ for some θ, $p=2,\ s=3$, fourth order (new ANTARES).

θ	$g(\theta)$
$-\pi$	$\frac{(-3+2\,\mu)^4}{(3+2\,\mu)^4}$
$-\frac{3\pi}{4}$	$\frac{\left(-48+15\,\mu+8\,\mu\,\sqrt{2}\right)^4}{\left(48+15\,\mu+8\,\mu\,\sqrt{2}\right)^4}$
$-\frac{\pi}{2}$	$\frac{(-24+7\,\mu)^4}{(24+7\,\mu)^4}$
$-\frac{\pi}{4}$	$\frac{\left(48-15\,\mu+8\,\mu\,\sqrt{2}\right)^4}{\left(-48-15\,\mu+8\,\mu\,\sqrt{2}\right)^4}$
0	1
$\frac{\pi}{4}$	$\frac{\left(48-15\,\mu+8\,\mu\,\sqrt{2}\right)^4}{\left(-48-15\,\mu+8\,\mu\,\sqrt{2}\right)^4}$
$\frac{\pi}{2}$	$\frac{(-24+7\,\mu)^4}{(24+7\,\mu)^4}$
$\frac{3\pi}{4}$	$\frac{\left(-48+15\,\mu+8\,\mu\,\sqrt{2}\right)^4}{\left(48+15\,\mu+8\,\mu\,\sqrt{2}\right)^4}$
π	$\frac{(-3+2\,\mu)^4}{(3+2\,\mu)^4}$

Table 3.25: Values of $g(\theta)$ for some θ, $p=2,\ s=4$, fourth order (new ANTARES).

θ	$g(\theta)$
$-\pi$	$-\frac{\left(1+\sqrt{3}\right)\left(-27+48\,\mu\,\sqrt{3}-256\,\mu^2-27\,\sqrt{3}+144\,\mu\right)}{\left(9+16\,\mu+3\,\sqrt{3}\right)^2}$
$-\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-108+90\,\mu\,\sqrt{3}+48\,\mu\,\sqrt{3}\sqrt{2}-353\,\mu^2-240\,\mu^2\sqrt{2}-108\,\sqrt{3}+270\,\mu+144\,\mu\,\sqrt{2}\right)}{\left(18+15\,\mu+8\,\mu\,\sqrt{2}+6\,\sqrt{3}\right)^2}$
$-\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-27+21\,\mu\,\sqrt{3}-49\,\mu^2-27\,\sqrt{3}+63\,\mu\right)}{\left(9+7\,\mu+3\,\sqrt{3}\right)^2}$
$-\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(108-90\,\mu\,\sqrt{3}+48\,\mu\,\sqrt{3}\sqrt{2}+353\,\mu^2-240\,\mu^2\sqrt{2}+108\,\sqrt{3}-270\,\mu+144\,\mu\,\sqrt{2}\right)}{\left(18+15\,\mu-8\,\mu\,\sqrt{2}+6\,\sqrt{3}\right)^2}$
0	1
$\frac{\pi}{4}$	$\frac{\left(1+\sqrt{3}\right)\left(108-90\,\mu\,\sqrt{3}+48\,\mu\,\sqrt{3}\sqrt{2}+353\,\mu^2-240\,\mu^2\sqrt{2}+108\,\sqrt{3}-270\,\mu+144\,\mu\,\sqrt{2}\right)}{\left(18+15\,\mu-8\,\mu\,\sqrt{2}+6\,\sqrt{3}\right)^2}$
$\frac{\pi}{2}$	$-\frac{\left(1+\sqrt{3}\right)\left(-27+21\,\mu\,\sqrt{3}-49\,\mu^2-27\,\sqrt{3}+63\,\mu\right)}{\left(9+7\,\mu+3\,\sqrt{3}\right)^2}$
$\frac{3\pi}{4}$	$-\frac{\left(1+\sqrt{3}\right)\left(-108+90\,\mu\,\sqrt{3}+48\,\mu\,\sqrt{3}\sqrt{2}-353\,\mu^2-240\,\mu^2\sqrt{2}-108\,\sqrt{3}+270\,\mu+144\,\mu\,\sqrt{2}\right)}{\left(18+15\,\mu+8\,\mu\,\sqrt{2}+6\,\sqrt{3}\right)^2}$
π	$-\frac{\left(1+\sqrt{3}\right)\left(-27+48\,\mu\,\sqrt{3}-256\,\mu^2-27\,\sqrt{3}+144\,\mu\right)}{\left(9+16\,\mu+3\,\sqrt{3}\right)^2}$

Table 3.26: Values of $g(\theta)$ for some θ, $p=3,\ s=2$, fourth order (new ANTARES).

θ	$g(\theta)$
$-\pi$	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(1782-4968\,\mu-3888\,\mu\sqrt{2}+6336\,\mu^2-4352\,\mu^3+1215\sqrt{2}+4320\sqrt{2}\mu^2\right)}{\left(6+8\,\mu+3\sqrt{2}\right)^3}$
$-\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-228096+546912\,\mu+392256\,\mu\sqrt{2}-538776\,\mu^2-380700\sqrt{2}\mu^2+155295\,\mu^3+109208\,\mu^3\sqrt{2}-155520\sqrt{2}\right)}{\left(24+15\,\mu+8\,\mu\sqrt{2}+12\sqrt{2}\right)^3}$
$-\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(28512-34776\,\mu-27216\,\mu\sqrt{2}+19404\,\mu^2-5831\,\mu^3+19440\sqrt{2}+13230\sqrt{2}\mu^2\right)}{\left(12+7\,\mu+6\sqrt{2}\right)^3}$
$-\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(228096-49248\,\mu-74304\,\mu\sqrt{2}+20376\,\mu^2+540\sqrt{2}\mu^2-155295\,\mu^3+109208\,\mu^3\sqrt{2}+155520\sqrt{2}\right)}{\left(-24-15\,\mu+8\,\mu\sqrt{2}-12\sqrt{2}\right)^3}$
0	1
$\frac{\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(228096-49248\,\mu-74304\,\mu\sqrt{2}+20376\,\mu^2+540\sqrt{2}\mu^2-155295\,\mu^3+109208\,\mu^3\sqrt{2}+155520\sqrt{2}\right)}{\left(-24-15\,\mu+8\,\mu\sqrt{2}-12\sqrt{2}\right)^3}$
$\frac{\pi}{2}$	$\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(28512-34776\,\mu-27216\,\mu\sqrt{2}+19404\,\mu^2-5831\,\mu^3+19440\sqrt{2}+13230\sqrt{2}\mu^2\right)}{\left(12+7\,\mu+6\sqrt{2}\right)^3}$
$\frac{3\pi}{4}$	$-\frac{1}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(-228096+546912\,\mu+392256\,\mu\sqrt{2}-538776\,\mu^2-380700\sqrt{2}\mu^2+155295\,\mu^3+109208\,\mu^3\sqrt{2}-155520\sqrt{2}\right)}{\left(24+15\,\mu+8\,\mu\sqrt{2}+12\sqrt{2}\right)^3}$
π	$\frac{2}{51}\,\frac{\left(5+2\sqrt{2}\right)\left(1782-4968\,\mu-3888\,\mu\sqrt{2}+6336\,\mu^2-4352\,\mu^3+1215\sqrt{2}+4320\sqrt{2}\mu^2\right)}{\left(6+8\,\mu+3\sqrt{2}\right)^3}$

Table 3.27: Values of $g(\theta)$ for some θ, $p=3,\ s=3$, fourth order (new ANTARES).

In Table 3.27, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$. It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$.
$p=3$: With $\mu = b\frac{\Delta t}{\Delta x^2}$, we computed $g(\theta)$ by MAPLE for $s=4$.
In Table 3.28, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0,\ldots,8\}$, where the results are rounded to two significant digits: It appears that there are certain restrictions on μ if we require $|g(\theta)| < 1$.

θ	$g(\theta)$
$-\pi$	$-0.21\ \frac{4400000.0\,\mu^3-2400000.0-8600000.0\,\mu^2+7200000.0\,\mu-870000.0\,\mu^4}{(27.0+16.0\,\mu)^4}$
$-\frac{3\pi}{4}$	$-0.052\ \frac{-160000000.0-370000000.0\,\mu^2+160000000.0\,\mu^3-26000000.0\,\mu^4+380000000.0\,\mu}{(53.0+26.0\,\mu)^4}$
$-\frac{\pi}{2}$	$-0.052\ \frac{-10000000.0+12000000.0\,\mu-6700000.0\,\mu^2+1400000.0\,\mu^3-130000.0\,\mu^4}{(27.0+7.0\,\mu)^4}$
$-\frac{\pi}{4}$	$0.052\ \frac{5000000.0\,\mu^2+1000000.0\,\mu^3+160000000.0-59000000.0\,\mu}{(53.0+4.0\,\mu)^4}$
0	1
$\frac{\pi}{4}$	$0.052\ \frac{5000000.0\,\mu^2+1000000.0\,\mu^3+160000000.0-59000000.0\,\mu}{(53.0+4.0\,\mu)^4}$
$\frac{\pi}{2}$	$-0.052\ \frac{-10000000.0+12000000.0\,\mu-6700000.0\,\mu^2+1400000.0\,\mu^3-130000.0\,\mu^4}{(27.0+7.0\,\mu)^4}$
$\frac{3\pi}{4}$	$-0.052\ \frac{-160000000.0-370000000.0\,\mu^2+160000000.0\,\mu^3-26000000.0\,\mu^4+380000000.0\,\mu}{(53.0+26.0\,\mu)^4}$
π	$-0.21\ \frac{4400000.0\,\mu^3-2400000.0-8600000.0\,\mu^2+7200000.0\,\mu-870000.0\,\mu^4}{(27.0+16.0\,\mu)^4}$

Table 3.28: Values of $g(\theta)$ for some θ, $p = 3,\ s = 4$, fourth order (new ANTARES).

Chapter 4

Comparison with Osher/Shu methods

To assess the results derived for the SDIRK methods in Chapters 2 and 3, we give the analogous analysis for the methods from [38]. As the simplest one step scheme, we first of all discuss the forward Euler method.

4.1 Forward Euler method

The forward Euler method is characterized by the Butcher array

$$\begin{aligned} a_{1,1} &= 0, \\ b_1 &= 1. \end{aligned}$$

The stability function for this scheme is

$$R(z) = 1 + z.$$

Since this is a polynomial (as for any explicit Runge–Kutta method), the stability region is of course bounded. It is plotted in Figure 4.1. It is easily seen analytically that the left boundary of the stability region is located at $\Re(z) = -2$.

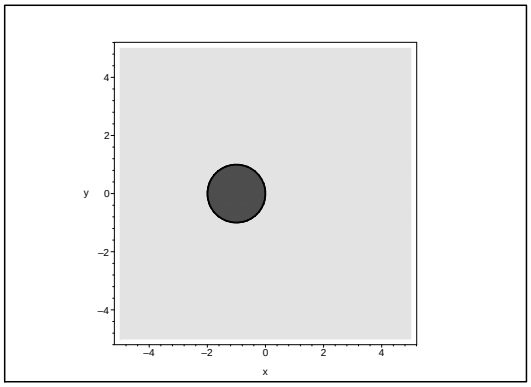


Figure 4.1: Stability region of the forward Euler method.

With a MATLAB implementation we check the theoretical order one of the method, and moreover, we observe the accuracy for a given stepsize analogously to Tables 2.3 and 2.5. The results are given in Table 4.1.
The dissipativity analysis for the three-point scheme (3.21) confirms that [43]

$$g(\theta) = 2\,\mu\,\cos(\theta) + 1 - 2\,\mu. \tag{4.1}$$

Δt	error	order	C
6.5000e−01	2.0275e+00		—
3.2500e−01	1.6771e+00	0.27	2.2812e+00
1.6250e−01	1.2750e+00	0.40	2.6157e+00
8.1250e−02	8.7896e−01	0.68	4.8276e+00
2.0313e−02	3.1573e−01	0.80	7.0891e+00
1.0156e−02	1.7108e−01	0.88	9.8917e+00
5.0781e−03	8.9353e−02	0.94	1.2623e+01
2.5391e−03	4.5705e−02	0.97	1.4793e+01
1.2695e−03	2.3120e−02	0.98	1.6281e+01

Table 4.1: Empirical convergence order for forward Euler method applied to (2.13).

θ	$g(\theta)$
$-\pi$	$1-4\,\mu$
$-\frac{3\pi}{4}$	$1-\mu\sqrt{2}-2\,\mu$
$-\frac{\pi}{2}$	$1-2\,\mu$
$-\frac{\pi}{4}$	$1+\mu\sqrt{2}-2\,\mu$
0	1
$\frac{\pi}{4}$	$1+\mu\sqrt{2}-2\,\mu$
$\frac{\pi}{2}$	$1-2\,\mu$
$\frac{3\pi}{4}$	$1-\mu\sqrt{2}-2\,\mu$
π	$1-4\,\mu$

Table 4.2: Values of $g(\theta)$ for forward Euler method, three-point differences.

In Table 4.2, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a linear function of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. It is readily confirmed that the results (3.6) and (3.7) are thus reproduced.
The dissipativity analysis for the scheme (3.40) shows that

$$g(\theta) \quad = \quad \frac{19}{12}\,\mu\,(\cos(\theta))^2 - 5/3\,\mu - 3/4\,\mu\,(\cos(\theta))^3 + 3/4\,\mu\,\cos(\theta) + 1/12\,\mu\,(\cos(\theta))^4 + 1. \tag{4.2}$$

In Table 4.3, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. $g(\pm 1) = 1$ holds irrespectively of μ.
The dissipativity analysis for the scheme (3.42) shows that

$$g(\theta) \quad = \quad 5/3\,\mu\,(\cos(\theta))^2 - \frac{16}{9}\,\mu - \frac{8}{9}\,\mu\,(\cos(\theta))^3 + \frac{8}{9}\,\mu\,\cos(\theta) + 1/9\,\mu\,(\cos(\theta))^4 + 1. \tag{4.3}$$

In Table 4.4, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. $g(\pm 1) = 1$ holds irrespectively of μ.
The dissipativity analysis for the scheme (3.44) yielded $g(\theta)$, computed by MAPLE:

$$g(\theta) \quad = \quad -1/3\,\mu\,(\cos(\theta))^2 - 7/3\,\mu + 8/3\,\mu\,\cos(\theta) + 1. \tag{4.4}$$

In Table 4.5, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ.

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1 - 3/16\,\mu\sqrt{2} - \frac{41}{48}\,\mu$
$-\frac{\pi}{2}$	$1 - 5/3\,\mu$
$-\frac{\pi}{4}$	$1 + 3/16\,\mu\sqrt{2} - \frac{41}{48}\,\mu$
0	1
$\frac{\pi}{4}$	$1 + 3/16\,\mu\sqrt{2} - \frac{41}{48}\,\mu$
$\frac{\pi}{2}$	$1 - 5/3\,\mu$
$\frac{3\pi}{4}$	$1 - 3/16\,\mu\sqrt{2} - \frac{41}{48}\,\mu$
π	1

Table 4.3: Values of $g(\theta)$ for forward Euler method, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1 - 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu$
$-\frac{\pi}{2}$	$1 - \frac{16}{9}\,\mu$
$-\frac{\pi}{4}$	$1 + 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu$
0	1
$\frac{\pi}{4}$	$1 + 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu$
$\frac{\pi}{2}$	$1 - \frac{16}{9}\,\mu$
$\frac{3\pi}{4}$	$1 - 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu$
π	1

Table 4.4: Values of $g(\theta)$ for forward Euler method, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	$1 - 16/3\,\mu$
$-\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2}$
$-\frac{\pi}{2}$	$1 - 7/3\,\mu$
$-\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2}$
$\frac{\pi}{2}$	$1 - 7/3\,\mu$
$\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2}$
π	$1 - 16/3\,\mu$

Table 4.5: Values of $g(\theta)$ for forward Euler method, fourth order (new ANTARES).

Δt	error	order	C
6.5000e−01	8.6275e−01	—	—
3.2500e−01	4.8773e−01	0.82	1.2298e+00
1.6250e−01	2.0597e−01	1.24	1.9735e+00
8.1250e−02	6.8206e−02	1.59	3.7333e+00
4.0625e−02	1.9418e−02	1.81	6.4532e+00
2.0313e−02	5.1376e−03	1.92	9.0536e+00
1.0156e−02	1.3171e−03	1.96	1.0812e+01
5.0781e−03	3.3310e−04	1.98	1.1827e+01
2.5391e−03	8.3736e−05	1.99	1.2385e+01
1.2695e−03	2.0990e−05	2.00	1.2691e+01

Table 4.6: Empirical convergence order for Osher/Shu 2 applied to (2.13).

4.2 Osher/Shu method of order 2

We now give the results for the second-order scheme [38, (2.16)], which is also commonly known as Heun's method. This is characterized by the Butcher array

$$\begin{aligned} &a_{2,1} = 1, \\ &a_{i,j} = 0 \quad \text{otherwise, } 1 \le i, j \le 2, \\ &b_1 = b_2 = \frac{1}{2}. \end{aligned}$$

The stability function for this scheme is

$$R(z) = 1 + z + \frac{z^2}{2}.$$

Since this is a polynomial (as for any explicit Runge–Kutta method), the stability region is of course bounded. It is plotted in Figure 4.2. It is easily seen analytically that the left boundary of the stability region is located at $\Re(z) = -2$.

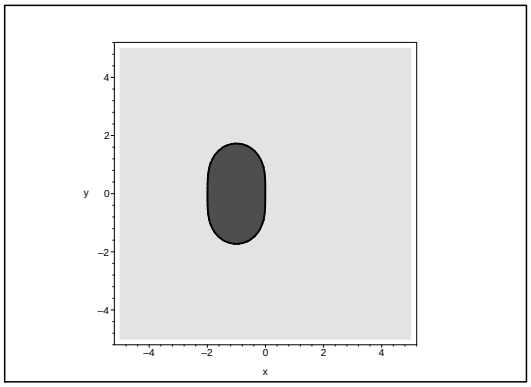


Figure 4.2: Stability region of second order Osher/Shu method.

With a MATLAB implementation we check the claimed order two of the method, and moreover, we compare the accuracy for a given stepsize with that for SDIRK 2 with $s = 1$ and $s = 6$ analogously to Tables 2.3 and 2.5. The results are given in Table 4.6.

θ	$g(\theta)$
$-\pi$	$1-4\,\mu+8\,\mu^2$
$-\frac{3\pi}{4}$	$1-\mu\sqrt{2}-2\,\mu+3\,\mu^2+2\,\mu^2\sqrt{2}$
$-\frac{\pi}{2}$	$2\,\mu^2-2\,\mu+1$
$-\frac{\pi}{4}$	$1+\mu\sqrt{2}-2\,\mu+3\,\mu^2-2\,\mu^2\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1+\mu\sqrt{2}-2\,\mu+3\,\mu^2-2\,\mu^2\sqrt{2}$
$\frac{\pi}{2}$	$2\,\mu^2-2\,\mu+1$
$\frac{3\pi}{4}$	$1-\mu\sqrt{2}-2\,\mu+3\,\mu^2+2\,\mu^2\sqrt{2}$
π	$1-4\,\mu+8\,\mu^2$

Table 4.7: Values of $g(\theta)$ for Osher/Shu method of order two, three-point differences.

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1-3/16\,\mu\sqrt{2}-\frac{41}{48}\,\mu+\frac{1843}{4608}\,\mu^2+\frac{41}{256}\,\mu^2\sqrt{2}$
$-\frac{\pi}{2}$	$1-5/3\,\mu+\frac{25}{18}\,\mu^2$
$-\frac{\pi}{4}$	$1+3/16\,\mu\sqrt{2}-\frac{41}{48}\,\mu+\frac{1843}{4608}\,\mu^2-\frac{41}{256}\,\mu^2\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1+3/16\,\mu\sqrt{2}-\frac{41}{48}\,\mu+\frac{1843}{4608}\,\mu^2-\frac{41}{256}\,\mu^2\sqrt{2}$
$\frac{\pi}{2}$	$1-5/3\,\mu+\frac{25}{18}\,\mu^2$
$\frac{3\pi}{4}$	$1-3/16\,\mu\sqrt{2}-\frac{41}{48}\,\mu+\frac{1843}{4608}\,\mu^2+\frac{41}{256}\,\mu^2\sqrt{2}$
π	1

Table 4.8: Values of $g(\theta)$ for Osher/Shu method of order two, fourth order (old ANTARES 'version 1').

The dissipativity analysis for the three-point scheme (3.21) shows that

$$g(\theta)=2\,\mu^2\left(\cos\left(\theta\right)\right)^2+2\,\mu^2+2\,\cos\left(\theta\right)\mu-4\,\cos\left(\theta\right)\mu^2+1-2\,\mu. \tag{4.5}$$

In Table 4.7, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)|<1$ can only hold under some bound on μ.
The dissipativity analysis for the scheme (3.40) shows that

$$\begin{aligned} g(\theta) &= 1-\frac{679}{288}\,\mu^2\left(\cos\left(\theta\right)\right)^2+\frac{19}{12}\,\mu\left(\cos\left(\theta\right)\right)^2-3/4\,\mu\left(\cos\left(\theta\right)\right)^3+3/4\,\mu\cos\left(\theta\right)+\frac{39}{16}\,\mu^2\left(\cos\left(\theta\right)\right)^3-\cdots\\ &\quad -5/4\,\mu^2\cos\left(\theta\right)+1/12\,\mu\left(\cos\left(\theta\right)\right)^4+\frac{53}{96}\,\mu^2\left(\cos\left(\theta\right)\right)^4-\frac{9}{8}\,\mu^2\left(\cos\left(\theta\right)\right)^5+\cdots\\ &\quad +\frac{119}{288}\,\mu^2\left(\cos\left(\theta\right)\right)^6-1/16\,\mu^2\left(\cos\left(\theta\right)\right)^7-5/3\,\mu+\frac{25}{18}\,\mu^2+\frac{1}{288}\,\mu^2\left(\cos\left(\theta\right)\right)^8. \end{aligned} \tag{4.6}$$

In Table 4.8, we evaluate the function $g(\theta)$ at the points $\theta\in\{-\pi+\ell\pi/4:\ell=0,\ldots,8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)|<1$ can only hold under some bound on μ. For $\theta\in\{-\pi,0,\pi\}$, $g(\theta)=1$ holds.
The dissipativity analysis for the scheme (3.42) shows that

$$\begin{aligned} g(\theta) &= 1-\frac{16}{9}\,\mu+\frac{128}{81}\,\mu^2-\frac{208}{81}\,\mu^2\left(\cos\left(\theta\right)\right)^2+5/3\,\mu\left(\cos\left(\theta\right)\right)^2+\frac{248}{81}\,\mu^2\left(\cos\left(\theta\right)\right)^3-\frac{128}{81}\,\mu^2\cos\left(\theta\right)-\frac{8}{9}\,\mu\left(\cos\left(\theta\right)\right)^3+\cdots\\ &\quad +\frac{8}{9}\,\mu\cos\left(\theta\right)+\frac{65}{162}\,\mu^2\left(\cos\left(\theta\right)\right)^4+1/9\,\mu\left(\cos\left(\theta\right)\right)^4-\frac{112}{81}\,\mu^2\left(\cos\left(\theta\right)\right)^5+\frac{47}{81}\,\mu^2\left(\cos\left(\theta\right)\right)^6-\frac{8}{81}\,\mu^2\left(\cos\left(\theta\right)\right)^7+\cdots\\ &\quad +\frac{1}{162}\,\mu^2\left(\cos\left(\theta\right)\right)^8. \end{aligned} \tag{4.7}$$

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1 - 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu + \frac{1217}{2592}\,\mu^2 + \frac{11}{54}\,\mu^2\sqrt{2}$
$-\frac{\pi}{2}$	$1 - \frac{16}{9}\,\mu + \frac{128}{81}\,\mu^2$
$-\frac{\pi}{4}$	$1 + 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu + \frac{1217}{2592}\,\mu^2 - \frac{11}{54}\,\mu^2\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1 + 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu + \frac{1217}{2592}\,\mu^2 - \frac{11}{54}\,\mu^2\sqrt{2}$
$\frac{\pi}{2}$	$1 - \frac{16}{9}\,\mu + \frac{128}{81}\,\mu^2$
$\frac{3\pi}{4}$	$1 - 2/9\,\mu\sqrt{2} - \frac{11}{12}\,\mu + \frac{1217}{2592}\,\mu^2 + \frac{11}{54}\,\mu^2\sqrt{2}$
π	1

Table 4.9: Values of $g(\theta)$ for Osher/Shu method of order two, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	$1 - 16/3\,\mu + \frac{128}{9}\,\mu^2$
$-\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 + 10/3\,\mu^2\sqrt{2}$
$-\frac{\pi}{2}$	$1 - 7/3\,\mu + \frac{49}{18}\,\mu^2$
$-\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 - 10/3\,\mu^2\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 - 10/3\,\mu^2\sqrt{2}$
$\frac{\pi}{2}$	$1 - 7/3\,\mu + \frac{49}{18}\,\mu^2$
$\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 + 10/3\,\mu^2\sqrt{2}$
π	$1 - 16/3\,\mu + \frac{128}{9}\,\mu^2$

Table 4.10: Values of $g(\theta)$ for Osher/Shu method of order two, fourth order (new ANTARES).

In Table 4.9, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. For $\theta \in \{-\pi, 0, \pi\}$, $g(\theta) = 1$ holds.
The dissipativity analysis for the scheme (3.44) yielded $g(\theta)$, computed by MAPLE.
In Table 4.10, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ.

4.3 Osher/Shu method of order 3

Next, we discuss [38, (2.18)]. This is characterized by the Butcher array

$$
\begin{aligned}
&a_{2,1} = 1,\\
&a_{3,1} = a_{3,2} = \frac{1}{4},\\
&a_{i,j} = 0 \quad \text{otherwise, } 1 \le i, j \le 3,\\
&b_1 = b_2 = \frac{1}{6}, \quad b_3 = \frac{2}{3}.
\end{aligned}
$$

The stability function for this scheme is

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6}.$$

Since this is a polynomial (as for any explicit Runge–Kutta method), the stability region is of course bounded. It is plotted in Figure 4.3. The left boundary of the stability region was numerically determined to be located at $\Re(z) \approx -2.51$.

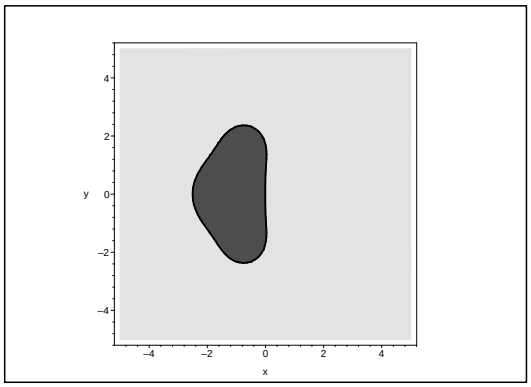


Figure 4.3: Stability region of third order Osher/Shu method.

Δt	error	order	C
6.5000e−01	5.2044e−01	—	—
3.2500e−01	2.0461e−01	1.35	9.2972e−01
1.6250e−01	5.1040e−02	2.00	1.9439e+00
8.1250e−02	9.1796e−03	2.48	4.5831e+00
4.0625e−02	1.3687e−03	2.75	9.0364e+00
2.0313e−02	1.8615e−04	2.88	1.3826e+01
1.0156e−02	2.4239e−05	2.94	1.7652e+01
5.0781e−03	3.0913e−06	2.97	2.0258e+01
2.5391e−03	3.9027e−07	2.99	2.1884e+01
1.2695e−03	4.9026e−08	2.99	2.2845e+01

Table 4.11: Empirical convergence order for Osher/Shu 3 applied to (2.13).

With a MATLAB implementation we check the claimed order three of the method, and moreover, we compare the accuracy for a given stepsize with that for SDIRK 3 with $s = 2$ and $s = 6$ analogously to Tables 2.5 and 2.6. The results are given in Table 4.11.
The dissipativity analysis for the three-point scheme (3.21) shows that

$$g(\theta) = -4\,\mu^3\,(\cos(\theta))^2 - 4/3\,\mu^3 + 2\,\mu^2\,(\cos(\theta))^2 + 2\,\mu^2 + 4/3\,\mu^3\,(\cos(\theta))^3 + 4\,\mu^3\cos(\theta) - 4\,\cos(\theta)\,\mu^2 + 2\,\cos(\theta)\,\mu + 1 - 2\,\mu. \quad (4.8)$$

In Table 4.12, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ.
For the dissipativity analysis for the scheme (3.40), we computed $g(\theta)$ by Maple.
In Table 4.13, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. For $\theta \in \{-\pi, 0, \pi\}$, $g(\theta) = 1$ holds.
For the dissipativity analysis for the scheme (3.42) we computed $g(\theta)$ by Maple.
In Table 4.14, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ. For $\theta \in \{-\pi, 0, \pi\}$, $g(\theta) = 1$ holds.
For the dissipativity analysis for the scheme (3.44) we computed $g(\theta)$ by Maple.
In Table 4.15, we evaluate the function $g(\theta)$ at the points $\theta \in \{-\pi + \ell\pi/4 : \ell = 0, \ldots, 8\}$. Since $g(\theta)$ is a polynomial of the variable μ, obviously $|g(\theta)| < 1$ can only hold under some bound on μ.

θ	$g(\theta)$
$-\pi$	$1-4\,\mu+8\,\mu^2-\frac{32}{3}\,\mu^3$
$-\frac{3\pi}{4}$	$1-\mu\,\sqrt{2}-2\,\mu+3\,\mu^2+2\,\mu^2\sqrt{2}-7/3\,\mu^3\sqrt{2}-10/3\,\mu^3$
$-\frac{\pi}{2}$	$1-2\,\mu+2\,\mu^2-4/3\,\mu^3$
$-\frac{\pi}{4}$	$1+\mu\,\sqrt{2}-2\,\mu+3\,\mu^2-2\,\mu^2\sqrt{2}+7/3\,\mu^3\sqrt{2}-10/3\,\mu^3$
0	1
$\frac{\pi}{4}$	$1+\mu\,\sqrt{2}-2\,\mu+3\,\mu^2-2\,\mu^2\sqrt{2}+7/3\,\mu^3\sqrt{2}-10/3\,\mu^3$
$\frac{\pi}{2}$	$1-2\,\mu+2\,\mu^2-4/3\,\mu^3$
$\frac{3\pi}{4}$	$1-\mu\,\sqrt{2}-2\,\mu+3\,\mu^2+2\,\mu^2\sqrt{2}-7/3\,\mu^3\sqrt{2}-10/3\,\mu^3$
π	$1-4\,\mu+8\,\mu^2-\frac{32}{3}\,\mu^3$

Table 4.12: Values of $g(\theta)$ for Osher/Shu method of order three.

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1-2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2+\frac{11}{54}\,\mu^2\sqrt{2}-\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
$-\frac{\pi}{2}$	$1-\frac{16}{9}\,\mu+\frac{128}{81}\,\mu^2-\frac{2048}{2187}\,\mu^3$
$-\frac{\pi}{4}$	$1+2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2-\frac{11}{54}\,\mu^2\sqrt{2}+\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
0	1
$\frac{\pi}{4}$	$1+2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2-\frac{11}{54}\,\mu^2\sqrt{2}+\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
$\frac{\pi}{2}$	$1-\frac{16}{9}\,\mu+\frac{128}{81}\,\mu^2-\frac{2048}{2187}\,\mu^3$
$\frac{3\pi}{4}$	$1-2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2+\frac{11}{54}\,\mu^2\sqrt{2}-\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
π	1

Table 4.13: Values of $g(\theta)$ for Osher/Shu method of order three, fourth order (old ANTARES 'version 1').

θ	$g(\theta)$
$-\pi$	1
$-\frac{3\pi}{4}$	$1-2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2+\frac{11}{54}\,\mu^2\sqrt{2}-\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
$-\frac{\pi}{2}$	$1-\frac{16}{9}\,\mu+\frac{128}{81}\,\mu^2-\frac{2048}{2187}\,\mu^3$
$-\frac{\pi}{4}$	$1+2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2-\frac{11}{54}\,\mu^2\sqrt{2}+\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
0	1
$\frac{\pi}{4}$	$1+2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2-\frac{11}{54}\,\mu^2\sqrt{2}+\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
$\frac{\pi}{2}$	$1-\frac{16}{9}\,\mu+\frac{128}{81}\,\mu^2-\frac{2048}{2187}\,\mu^3$
$\frac{3\pi}{4}$	$1-2/9\,\mu\,\sqrt{2}-\frac{11}{12}\,\mu+\frac{1217}{2592}\,\mu^2+\frac{11}{54}\,\mu^2\sqrt{2}-\frac{3395}{34992}\,\mu^3\sqrt{2}-\frac{5401}{31104}\,\mu^3$
π	1

Table 4.14: Values of $g(\theta)$ for Osher/Shu method of order three, fourth order (old ANTARES 'version 2').

θ	$g(\theta)$
$-\pi$	$1 - 16/3\,\mu + \frac{128}{9}\,\mu^2 - \frac{2048}{81}\,\mu^3$
$-\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 + 10/3\,\mu^2\sqrt{2} - \frac{1015}{144}\,\mu^3 - \frac{803}{162}\,\mu^3\sqrt{2}$
$-\frac{\pi}{2}$	$1 - 7/3\,\mu + \frac{49}{18}\,\mu^2 - \frac{343}{162}\,\mu^3$
$-\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 - 10/3\,\mu^2\sqrt{2} - \frac{1015}{144}\,\mu^3 + \frac{803}{162}\,\mu^3\sqrt{2}$
0	1
$\frac{\pi}{4}$	$1 - 5/2\,\mu + 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 - 10/3\,\mu^2\sqrt{2} - \frac{1015}{144}\,\mu^3 + \frac{803}{162}\,\mu^3\sqrt{2}$
$\frac{\pi}{2}$	$1 - 7/3\,\mu + \frac{49}{18}\,\mu^2 - \frac{343}{162}\,\mu^3$
$\frac{3\pi}{4}$	$1 - 5/2\,\mu - 4/3\,\mu\sqrt{2} + \frac{353}{72}\,\mu^2 + 10/3\,\mu^2\sqrt{2} - \frac{1015}{144}\,\mu^3 - \frac{803}{162}\,\mu^3\sqrt{2}$
π	$1 - 16/3\,\mu + \frac{128}{9}\,\mu^2 - \frac{2048}{81}\,\mu^3$

Table 4.15: Values of $g(\theta)$ for Osher/Shu method of order three, fourth order (new ANTARES).

	Δt	err	fcount
Explicit Euler	8.650e−8	9.999e−5	17572000
O/S 2	1.711e−4	9.996e−5	17770
O/S 3	1.612e−3	9.986e−5	2829
SDIRK $p = 2,\ s = 2$	1.040e−3	9.968e−5	4816
SDIRK $p = 2,\ s = 3$	1.619e−3	9.999e−5	4208
SDIRK $p = 2,\ s = 4$	2.190e−3	9.918e−5	3934
SDIRK $p = 2,\ s = 5$	2.764e−3	9.860e−5	3770
SDIRK $p = 2,\ s = 6$	3.341e−3	9.873e−5	3659
SDIRK $p = 3,\ s = 2$	4.318e−3	9.901e−5	1729
SDIRK $p = 3,\ s = 3$	5.892e−3	9.954e−5	1765
SDIRK $p = 3,\ s = 4$	7.238e−3	9.937e−5	1829
SDIRK $p = 3,\ s = 5$	8.444e−3	9.835e−5	1890
SDIRK $p = 3,\ s = 6$	9.620e−3	9.918e−5	1950

Table 4.16: Step size Δt, error at $t = 1.52$ and number of evaluations of the right-hand side, comparison of SDIRK and Osher/Shu methods when applied to (2.13).

4.4 Comparison of the efficiency

To conclude the assessment of the merits of the SDIRK methods as compared to the forward Euler and Osher/Shu methods, we make the following comparison: For the test problem (2.13), we manually adjust the stepsizes 'Δt' for all the investigated methods such that the errors 'err' at $t = 1.52$ are just below 10^{-4}, and count the number of evaluations of the right-hand side 'fcount' necessary to achieve this precision. We noticed a critical dependence of the performance of the SDIRK methods on the tolerance for the fixed point iteration. Table 4.16 gives the results where the tolerance for the fixed point iteration was chosen as 10^{-5} for SDIRK with $p = 2$ and 10^{-6} for SDIRK with $p = 3$. The investigations reported in Tables 2.11 and 2.12 in Chapter 2 demonstrate that the performance of the SDIRK methods could be further enhanced by careful tuning of this parameter. We notice that SDIRK 3 requires the fewest evaluations, but the number increases with the number s of stages in that case. Osher/Shu of order $p = 3$ is the second best choice, and the performance of SDIRK 2 increases with s. The forward Euler method and Osher/Shu of order two show an unacceptable performance.

Chapter 5

Conclusions

We have analyzed the numerical properties of discretization methods which are employed in the ANTARES code. For time integration, we put forward the use of *singly diagonally implicit Runge–Kutta (SDIRK) schemes* of orders two and three, respectively, which are *total variation diminishing* with large step-size coefficients. These can be implemented with a memory requirement which is independent of the number of stages. Due to the construction, the steps performed in each of the s stages of the second-order SDIRK method resemble the *trapezoidal rule*, and indeed the methods are A-stable. However, the integration performed at the end of each sub-interval implies that the SDIRK method yields a different approximation. This is also reflected in the fact that the dissipativity analysis yields a different behavior for s even or s odd: The analysis summed up in Table 5.2 below shows that for s odd, the amplification factor undergoes a sign change which yields oscillatory solution behavior, while it is nonnegative for s even, implying monotonic behavior. The restarts at the ends of the Runge–Kutta sub-intervals also imply that the computation of the first stage corresponds to a step of length $h/2s$, while the step-length would correspond to h/s indiscriminately for a composition of steps of the trapezoidal rule. Moreover, fewer fixed point iterations are needed to reach a prescribed tolerance when s is higher, and the error constants are smaller for larger s, a significant improvement over the explicit Osher/Shu methods.
The third-order SDIRK methods have bounded stability regions which grow with the number of stages. The error constants are smaller than the one for the third-order explicit Osher/Shu method, and also smaller than for SDIRK 2. This implies an advantageous behavior of the fixed point iteration used for the solution of the non-linear algebraic equations arising for each stage. However, the radius of convergence of the fixed point iteration limits the step-sizes which can maximally be used. By a heuristic reasoning we determined that, relating the radius of convergence of the fixed point iteration to the restrictions for stability for both the diffusive and the advective parts, the efficiency of the implicit SDIRK methods of second order may be at most comparable to the explicit methods, while there may be some moderate gain for the third order methods, assuming that only two fixed point iterations are required for each stage in the second order case and three iterations for the third order methods. The major advantage in each case is that the SDIRK methods have significantly smaller error constants than the explicit methods, whence we can expect more accurate results with comparable computational effort. This fact is also reflected in the experiments reported in Table 4.16. We found for an ODE test problem that the computational effort to reach the same accuracy, taking into account the additional work introduced by the fixed point iteration, for the third order SDIRK methods was the least, with $p = 3,\ s = 2$ giving the most efficient method, while for $p = 2$, higher s is advantageous. The explicit and implicit methods of third order excelled over the second order methods, and the implicit SDIRK methods always performed better than their explicit counterparts of the same orders.
Moreover, we have analyzed the dissipativity of the spatial and time discretisations discussed in this report. We summarize the results in Table 5.1. For the ten different time integrators we discussed in detail, and the four space discretisations, we indicate for which θ the condition $|g(\theta)| < 1$ is violated. We observed the following different types of behavior, indicated in Table 5.1 by the associated integers $1, \ldots, 6$:

1 ... indicates that the condition $|g(\theta)| < 1$ is satisfied for all θ considered.

2 ... indicates that the condition $|g(\theta)| < 1$ is violated for $\theta = \pm\pi$, and holds otherwise.

3 ... indicates that the condition $|g(\theta)| < 1$ is violated for $\theta = \pm\pi$, the validity is unknown otherwise.

Integrator / space discretisation	3pt	(3.40)	(3.42)	(3.44)
SDIRK, $p = 2,\ s = 1$	1	2	2	1
SDIRK, $p = 2,\ s = 2$	1	2	2	1
SDIRK, $p = 2,\ s = 3$	1	2	2	1
SDIRK, $p = 2,\ s = 4$	1	2	2	1
SDIRK, $p = 3,\ s = 2$	6	3	3	6
SDIRK, $p = 3,\ s = 3$	6	3	3	6
SDIRK, $p = 3,\ s = 4$	6	3	3	6
Explicit Euler	4	5	5	4
Osher/Shu, $p = 2$	4	5	5	4
Osher/Shu, $p = 3$	4	5	5	4

Table 5.1: Summary of dissipativity analysis

4 ... indicates that the condition $|g(\theta)| < 1$ holds for $\mu \leq \mathcal{C}$ with some finite constant $\mathcal{C}$. This restriction is obvious for the explicit methods, because $g(\theta)$ is a polynomial of μ.

5 ... indicates that the condition $|g(\theta)| < 1$ holds for $\mu \leq \mathcal{C}$ with some finite constant $\mathcal{C}$ for $\theta \neq \pm\pi$, but is violated for $\theta = \pm\pi$.

6 ... unknown.

From the dissipativity analysis, we can draw further conclusions. To this end, we compute the first positive zero of the function $g(\mu, \pi)$ (note that the results are the same for $g(\mu, -\pi)$ since all the space discretisations we investigated are symmetric). The results are given in Table 5.2. For each time integrator and space discretisation, we give in the first column the first positive root with change of sign, in the second the first positive root, and in the third the smallest positive point where $|g(\mu, -\pi)|$ becomes larger than or equal to 1. The results are rounded to four significant digits. For the discretisations (3.40) and (3.42), $g(\pm\pi) = 1$, and hence there is no damping on the grid scale, whence no conclusions on the step-size choice can be drawn from the dissipativity analysis. For the dissipative space discretisations, the three-point scheme (3.21) and (3.44), we observe that the first positive root of g grows when the number of stages s in the time integrator is chosen larger, and that the solution behavior is monotonic for s even, while oscillatory behavior may be observed for s odd even when the method is still dissipative. For $p = 2$, the modulus of the amplification factor is always smaller than 1, while for $p = 3$ this requirement implies a restriction. The limitations for the fourth-order dissipative spatial discretisation are more restrictive than those for the three-point scheme by a factor 3/4, which confirms the theoretical considerations from Section 3.2.

Integrator / space discretisation	3pt			(3.40)			(3.42)			(3.44)		
SDIRK, $p = 2,\ s = 1$	0.5	0.5	—	—	—	—	—	—	—	0.375	0.375	—
SDIRK, $p = 2,\ s = 2$	—	1	—	—	—	—	—	—	—	—	0.75	—
SDIRK, $p = 2,\ s = 3$	1.5	1.5	—	—	—	—	—	—	—	1.125	1.125	—
SDIRK, $p = 2,\ s = 4$	—	2.0	—	—	—	—	—	—	—	—	1.5	—
SDIRK, $p = 2,\ s = 5$	2.5	2.5	—	—	—	—	—	—	—	1.875	1.875	—
SDIRK, $p = 2,\ s = 6$	—	3.0	—	—	—	—	—	—	—	—	2.25	—
SDIRK, $p = 3,\ s = 2$	—	—	3.232	—	—	—	—	—	—	—	—	2.424
SDIRK, $p = 3,\ s = 3$	2.328	2.328	9.273	—	—	—	—	—	—	1.746	1.746	6.955
SDIRK, $p = 3,\ s = 4$	—	—	15.25	—	—	—	—	—	—	—	—	11.44
SDIRK, $p = 3,\ s = 5$	5.209	5.209	25.26	—	—	—	—	—	—	3.901	3.901	18.94
SDIRK, $p = 3,\ s = 6$	—	—	35.25	—	—	—	—	—	—	—	—	26.43
forward Euler	0.25	0.25	0.5	—	—	—	—	—	—	0.187	0.187	0.375
Osher/Shu, $p = 2$	—	—	0.5	—	—	—	—	—	—	—	—	0.375
Osher/Shu, $p = 3$	0.399	0.399	0.628	—	—	—	—	—	—	0.299	0.299	0.471

Table 5.2: Boundary of monotone solution behavior

Bibliography

[1] K. Dekker and J.G. Verwer. *Stability of Runge–Kutta Methods for Stiff Nonlinear Differential Equations*, volume 2 of *CWI Monographs*. North-Holland, Amsterdam, 1984.

[2] R. Donat and A. Marquina. Capturing shock reflections: An improved flux formula. *J. Comput. Phys.*, 125:42–58, 1996.

[3] L. Ferracina. *Monotonicity and Boundedness in General Runge–Kutta Methods*. Ph.D. Thesis, Leiden University, Leiden, The Netherlands, 2005.

[4] L. Ferracina and M. Spijker. An extension and analysis of the Shu–Osher representation of Runge–Kutta methods. *Math. Comp.*, 74:201–219, 2004.

[5] L. Ferracina and M. Spijker. Stepsize restrictions for total–variation–boundedness in general Runge–Kutta procedures. *SIAM J. Numer. Anal.*, 42:1073–1093, 2004.

[6] L. Ferracina and M. Spijker. Computing optimal monotonicity–preserving Runge–Kutta methods. Report no. MI 2005-07, Mathematical Institute, Leiden University, 2005.

[7] L. Ferracina and M. Spijker. Stepsize restrictions for total–variation–boundedness in general Runge–Kutta procedures. *Appl. Numer. Math.*, 53:265–279, 2005.

[8] L. Ferracina and M. Spijker. Strong stability of singly-diagonally-implicit Runge–Kutta methods. *Appl. Numer. Math.*, 58:1675–1686, 2008.

[9] B. Fornberg. *A Practical Guide to Pseudospectral Methods*. Cambridge Monographs on Applied and Computational Mathematics 1. Cambridge University Press, Cambridge, U.K., 1996.

[10] B. Freytag and M. Steffen. Numerical simulations of convection in A-stars. In J. Zverko, J. Ziznovsky, S.J. Adelman, and W.W. Weiss, editors, *The A-Star Puzzle (IAU Symp. 224)*, pages 139–147. Cambridge University Press, Cambridge, U.K., 2004.

[11] C.W. Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, N.J., 1971.

[12] S. Gottlieb and L.-A. Gottlieb. Strong stability preserving properties of Runge–Kutta time discretization methods for linear constant coefficient operators. *J. Sci. Comput.*, 18:83–109, 2003.

[13] S. Gottlieb, D. Ketcheson, and C.-W. Shu. High order strong stability preserving time discretizations. *J. Sci. Comput.*, 38:251–289, 2009.

[14] S. Gottlieb and S.J. Ruuth. Optimal strong-stability-preserving time-stepping schemes with fast downwind spatial discretizations. *J. Sci. Comput.*, 27:289–303, 2006.

[15] S. Gottlieb and C.-W. Shu. Total variation diminishing Runge–Kutta schemes. *Math. Comp.*, 67:73–85, 1998.

[16] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Rev.*, 43(1):89–112, 2001.

[17] R.D. Grigorieff. *Numerik Gewöhnlicher Differentialgleichungen*, volume 1. B.G. Teubner, Stuttgart, 1972.

[18] E. Hairer, S.P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I.* Springer-Verlag, Berlin–Heidelberg–New York, 1987.

[19] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II.* Springer-Verlag, Berlin–Heidelberg–New York, 1991.

[20] I. Higueras. Representations of Runge–Kutta methods and strong stability preserving methods. *SIAM J. Numer. Anal.*, 43:924–948, 2005.

[21] I. Higueras. Strong stability for additive Runge–Kutta methods. *SIAM J. Numer. Anal.*, 44:1735–1758, 2006.

[22] I. Higueras. Characterizing strong stability preserving additive Runge–Kutta methods. *J. Sci. Comput.*, 39:115–128, 2009.

[23] W. Hundsdorfer, A. Mozartova, and M. Spijker. Stepsize conditions for boundedness in numerical initial value problems. *SIAM J. Numer. Anal.*, 47(5):3797–3819, 2009.

[24] W. Hundsdorfer, A. Mozartova, and M. Spijker. Special boundedness properties in numerical initial value problems. Report ISSN 1386-3703, CWI Amsterdam, 2010.

[25] W. Hundsdorfer, S. Ruuth, and R. Spiteri. Monotonicity-preserving linear multistep methods. *SIAM J. Numer. Anal.*, 41(2):605–623, 2003.

[26] W. Hundsdorfer and M. Spijker. Boundedness and strong stability of Runge–Kutta methods. Report MI 2009-14, Mathematical Institute, Leiden University, 2009.

[27] D. Ketcheson. Highly efficient strong stability preserving Runge–Kutta methods with low storage implementations. *SIAM J. Sci. Comput.*, 30:2113–2136, 2008.

[28] D. Ketcheson, C. Macdonald, and S. Gottlieb. Optimal implicit strong stability preserving Runge–Kutta methods. *Appl. Numer. Math.*, 59:373–392, 2009.

[29] J.F.B.M. Kraaijevanger. Contractivity of Runge–Kutta methods. *BIT*, 31:482–528, 1991.

[30] F. Kupka, J. Ballot, and H.J. Muthsam. Effects of resolution and helium abundance in A star surface convection simulations. *Comm. in Asteroseismology*, 160:30–63, 2009.

[31] X. Liu and S. Osher. Convex ENO high order multi-dimensional schemes without field by field decomposition or staggered grids. *J. Comput. Phys.*, 142:304–330, 1998.

[32] H.J. Muthsam, F. Kupka, B. Löw-Baselli, C. Obertscheider, M. Langer, and P. Lenz. ANTARES — A Numerical Tool for Astrophysical RESearch with applications to solar granulation. *New Astronomy*, 15:460–475, 2010.

[33] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics.* Springer-Verlag, Berlin-Heidelberg, 2nd edition, 2007.

[34] S.J. Ruuth and R.J. Spiteri. Two barriers on strong-stability-preserving time discretization methods. *J. Sci. Comput.*, 17:211–220, 2002.

[35] C.-W. Shu. Total-variation-diminishing time discretizations. *SIAM J. Sci. Statist. Comput.*, 9:1073–1084, 1988.

[36] C.-W. Shu. Essentially non-oscillarory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. Technical Report ICASE 97-65, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA, 1997.

[37] C.-W. Shu. A survey of strong stability-preserving high-order time discretization methods. In *Collected Lectures on the Preservation of Stability under Discretization*, pages 51–65. SIAM, Philadelphia, PA, 2002.

[38] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock–capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.

[39] E.A. Spiegel. The smoothing of temperature fluctuations by radiative transfer. *Astrophys. Jour.*, 126:202–207, 1957.

[40] M. Spijker. Contractivity in the numerical solution of initial value problems. *Numer. Math.*, 42:271–290, 1983.

[41] M. Spijker. Stepsize conditions for general monotonicity in numerical initial value problems. *SIAM J. Numer. Anal.*, 45(3):1226–1245, 2007.

[42] R.J. Spiteri and S.J. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM J. Numer. Anal.*, 40:469–491, 2002.

[43] J.C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations*. SIAM, Philadelphia, PA, 2nd edition, 2004.

[44] J.S. Walker. *Fourier Analysis*. Oxford University Press, New York, 1988.

[45] A. Weiss, W. Hillebrandt, H.-C. Thomas, and H. Ritter. *Cox and Giuli's Principles of Stellar Structure*. Advances in Astronomy and Astrophysics. Cambridge Scientific Publishers Ltd., Cambridge, U.K., 2nd edition, 2004.

[46] D.V. Widder. *The Heat Equation*, volume 67 of *Pure and Applied Mathematics*. Academic Press, New York, 1975.

[47] J.H. Williamson. Low-storage Runge–Kutta schemes. *J. Comput. Phys.*, 35:48–56, 1980.