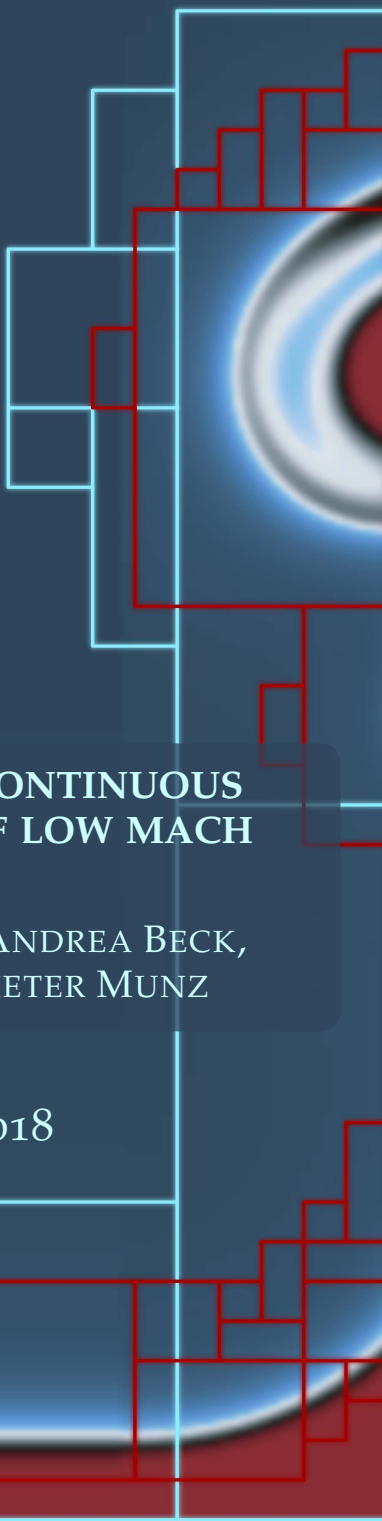
Communications in Applied Mathematics and Computational Science



EFFICIENT HIGH-ORDER DISCONTINUOUS GALERKIN COMPUTATIONS OF LOW MACH NUMBER FLOWS

JONAS ZEIFANG, KLAUS KAISER, ANDREA BECK, JOCHEN SCHÜTZ AND CLAUS-DIETER MUNZ

vol. 13 no. 2 2018

msp

COMM. APP. MATH. AND COMP. SCI.
Vol. 13, No. 2, 2018
dx.doi.org/10.2140/camcos.2018.13.243

EFFICIENT HIGH-ORDER DISCONTINUOUS GALERKIN COMPUTATIONS OF LOW MACH NUMBER FLOWS

JONAS ZEIFANG, KLAUS KAISER, ANDREA BECK,
JOCHEN SCHÜTZ AND CLAUS-DIETER MUNZ

We consider the efficient approximation of low Mach number flows by a high-order scheme, coupling a discontinuous Galerkin (DG) discretization in space with an implicit/explicit (IMEX) discretization in time. The splitting into linear implicit and nonlinear explicit parts relies heavily on the incompressible solution. The method has been originally developed for a singularly perturbed ODE and applied to the isentropic Euler equations. Here, we improve, extend, and investigate the so-called RS-IMEX splitting method. The resulting scheme can cope with a broader range of Mach numbers without running into roundoff errors, it is extended to realistic physical boundary conditions, and it is shown to be highly efficient in comparison to more standard solution techniques.

1. Introduction

Computing solutions to singularly perturbed problems can be cumbersome and expensive due to their multiscale nature. However, they do often occur in physical reality. A typical example is the transition from the compressible to the incompressible Navier–Stokes equations that constitutes a singularly perturbed limit [29; 41; 49]. Another example is multiphase flows, in which small liquid droplets are suspended in a gaseous phase. In such problems, the Mach number ε — the local ratio of flow speed to the speed of sound — can vary by orders of magnitude. In particular, some parts are very close to the incompressible regime, meaning that the Mach number is close to zero, while in other parts, ε is of the order of one. Flows of this nature are sometimes called *all-speed* flows.

Besides some issues such as the high-order treatment of the ubiquitous shocks or the treatment of turbulence, the efficient discretization of the $\varepsilon = \mathbb{O}(1)$ case is by now rather well understood; see, e.g., [47] and the references therein. In this work, we therefore focus on the efficient discretization of the $\varepsilon \ll 1$ case as a milestone towards *all-speed* schemes.

MSC2010: 35L65, 65N30.
Keywords: discontinuous Galerkin, IMEX-Runge–Kutta, low Mach number, splitting, asymptotic preserving.

The first idea that comes to mind is to treat the low Mach case as incompressible. In many situations however, compressible effects matter even in the vicinity of the incompressible regime. An example is given by the computation of transcritical droplets in a surrounding (supercritical) gas phase, where a strong coupling of thermodynamics and hydrodynamics in the droplets occurs. This phase state is also called the "compressible liquid" state and of current research interest [27; 26; 44]. Other situations occur in meteorological flows, where density gradients have to be considered but acoustic waves do not have to be resolved [14], and situations with strong temperature gradients but low velocities, e.g., natural convection [35]. As a consequence, we therefore stick to solution procedures for the compressible equations. The incompressible equations will nonetheless serve as a building block in our discretization.

Computing solutions to the low Mach equations ($\varepsilon \ll 1$) using classical discretization paradigms which mostly rely on *explicit* time stepping methods leads to the unwanted encounter of having to choose an extremely small time step size ($\Delta t \lesssim \varepsilon \Delta x$) to obtain a stable algorithm. Furthermore, due to the excessive amount of numerical viscosity that is added to the approximate solution, the explicit method may yield an incorrect solution [38]. One remedy is to use an implicit-explicit (IMEX) time stepping method [2; 28; 10] that relies on a splitting of the flux functions into a stiff part, which accounts for the singularity in the problem and is treated implicitly, and a nonstiff part, which only has a mild dependency on ε, and not on ε^{-1}. A number of splittings have been developed over the past few years; see, e.g., [30; 9; 13; 21]. All of those splittings have the disadvantage that it is very difficult to adapt them to other physical situations at hand, because they are developed for a specific set of equations.

To circumvent this problem, Kaiser et al. have introduced a general splitting and have applied it to the isentropic Euler equations in [25] that is based on the incompressible limit solution (called the *reference solution*); the splitting was hence termed *RS-IMEX*. The RS-IMEX idea is conceptually similar to the one introduced in [42] where the underlying problem is a singularly perturbed ODE. Related ideas have already been published earlier in [16; 9; 19]. One of the advantages of the splitting is that its idea, at least from a conceptual point of view, is independent from the underlying singularly perturbed problem and thereby not specific to a fixed system of equations. Furthermore, the implicit part is always linear in the solution variable, which usually reduces the time-to-solution tremendously, as the resulting algebraic equations are then also linear. Those linear equations are usually solved through a Krylov-type iteration method, which means that only matrix-vector products are needed, where the Jacobian is being frequently approximated via finite differences. For a nonlinear operator at low ε, this can pose severe problems for the approximation quality. However, as the implicit part of the RS-IMEX is linear,

the finite differences are not approximations but exact. In [24], the splitting has been used within a high-order IMEX discontinuous Galerkin (DG) solver and it has been shown that the algorithm preserves the asymptotics of the problem, which means that for $\varepsilon \to 0$, the discrete solution converges towards a discretization of the incompressible Euler equations. The latter is an important property as it means that no spurious effects stemming from the discretization and polluting the solution are introduced for small Mach numbers.

The purpose of this work is to improve, extend, and investigate the method introduced in [24] towards engineering problems.

- We improve the scheme for very small Mach numbers to suffer less from roundoff errors [33]. This is achieved through a reformulation of the method in a perturbed variable, partly following the lines of [43]. Thereby, we alleviate the dependency of the method on ε^{-1} to a great extend, which is the core source of roundoff problems.
- We extend the scheme to cope with more realistic physical settings by adding appropriate boundary conditions ([24] works with periodic ones) and considering three dimensions.
- We investigate the scheme with respect to runtime in the framework of a modern parallel architecture solver and compare it against other methods. For solving the algebraic system, we take advantage of the linearity of RS-IMEX in the solution process. We demonstrate the advantages of this novel method with respect to runtime and accuracy as a function of ε for nontrivial test cases.

The paper is structured as follows. In Section 2 we introduce the underlying isentropic Euler equations. Section 3 introduces the RS-IMEX splitting for those equations; subsequently, in Section 4, the discontinuous Galerkin discretization including the IMEX time discretization is detailed. In Section 5, we validate the method through a manufactured solution. Furthermore, we explain in detail how to circumvent a problem with machine accuracy due to low Mach numbers ε. In Section 6 we present more involved numerical examples and discussions concerning accuracy and efficiency in the low Mach number case. Finally, in Section 7 we offer conclusion and outlook.

2. Equations

In this work we consider the isentropic Euler equations on a domain $\Omega \subset \mathbb{R}^d$. Nondimensionalized, those equations are given by

$$\partial_t \boldsymbol{w} + \nabla_x \cdot \boldsymbol{F}(\boldsymbol{w}) = 0 \quad \text{with } \boldsymbol{w} := \begin{pmatrix} \rho \\ \rho \boldsymbol{u} \end{pmatrix} \text{ and } \boldsymbol{F}(\boldsymbol{w}) := \begin{pmatrix} \rho \boldsymbol{u} \\ \rho \boldsymbol{u} \otimes \boldsymbol{u} + (1/\varepsilon^2) p(\rho) \cdot \mathbf{Id} \end{pmatrix} \tag{1}$$

with $\boldsymbol{u}$ and ρ denoting velocity and density, respectively. The subscripts t and x denote temporal and spatial derivatives, respectively. The reference Mach number, defined as

$$\varepsilon := \frac{u^*}{\sqrt{p(\rho^*)/\rho^*}}$$

with u^* and ρ^* reference values used in the nondimensionalization process, is a measure for the compressibility of the system. The pressure p is defined by the equation of state

$$p(\rho) = \kappa \rho^\gamma, \tag{2}$$

with $\gamma \geq 1$ the isentropic coefficient and $\kappa > 0$ being a constant.

The eigenvalues of $\frac{\partial}{\partial \boldsymbol{w}} \boldsymbol{F}(\boldsymbol{w}) \cdot \boldsymbol{n}$ are, for $\Omega \subset \mathbb{R}^3$ and with $c := \sqrt{\gamma p/\rho}$ being the speed of sound, given by

$$\lambda_{1,2} = \boldsymbol{u} \cdot \boldsymbol{n}, \qquad \lambda_{3,4} = \boldsymbol{u} \cdot \boldsymbol{n} \pm \frac{c}{\varepsilon}. \tag{3}$$

It is obvious that for $\varepsilon \ll 1$ (the low Mach case) those waves have extremely different speeds, i.e., wave speeds are in $\mathbb{O}(1)$ and $\mathbb{O}(\varepsilon^{-1})$. In the limit $\varepsilon \to 0$ two wave speeds tend to infinity, meaning that the associated hyperbolic equation degenerates. This means that some information travels infinitely fast, and some at finite speed. Besides that, one can show that under certain conditions [29], the compressible equations (1) transform towards the incompressible Euler equations, which are given by

$$\begin{aligned} &\partial_t \begin{pmatrix} 0 & \\ & \mathbf{1} \end{pmatrix} \boldsymbol{w}_{(0)} + \nabla_x \cdot \boldsymbol{G}(\boldsymbol{w}_{(0)}, p_{(2)}) = 0 \quad \text{and} \quad \rho_{(0)} \equiv \text{const} \\ &\qquad \text{with} \qquad \boldsymbol{w}_{(0)} := \begin{pmatrix} \rho_{(0)} \\ (\rho \boldsymbol{u})_{(0)} \end{pmatrix} \\ &\qquad \text{and} \quad \boldsymbol{G}(\boldsymbol{w}_{(0)}, p_{(2)}) := \begin{pmatrix} (\rho \boldsymbol{u})_{(0)} \\ (\rho \boldsymbol{u})_{(0)} \otimes (\rho \boldsymbol{u})_{(0)}/\rho_{(0)} + p_{(2)} \cdot \mathbf{Id} \end{pmatrix}. \end{aligned} \tag{4}$$

The relation between the compressible and the incompressible equations can be understood best if we assume that every quantity of (1) can be represented by an asymptotic expansion, e.g.,

$$\rho = \rho_{(0)} + \varepsilon \rho_{(1)} + \varepsilon^2 \rho_{(2)} + \mathbb{O}(\varepsilon^3),$$

and compute the formal limit $\varepsilon \to 0$. The incompressible equations (4) are then obtained under the assumption of *well prepared* initial and boundary conditions; see, e.g., [21] for a detailed computation and [29] for a more formal proof.

Definition (well prepared initial and boundary conditions). We call initial conditions *well prepared* if they possess an asymptotic expansion in positive powers of ε,

and

$$\rho(t=0) = \text{const} + \mathbb{O}(\varepsilon^2) \quad \text{and} \quad \nabla_x \cdot (\rho \boldsymbol{u})(t=0) = \mathbb{O}(\varepsilon).$$

We call boundary conditions *well prepared* if they ensure

$$\int_{\partial\Omega} (\rho \boldsymbol{u})|_{\partial\Omega} \cdot \boldsymbol{n} = 0 \quad \text{and} \quad \rho|_{\partial\Omega} = \text{const} + \mathbb{O}(\varepsilon^2).$$

For *well prepared* initial conditions the corresponding incompressible state can be calculated by setting $\varepsilon = 0$. Consequently, incompressible density is the constant value $\rho_{(0)}$ and the velocity field is $\boldsymbol{u}_{(0)}$. To initialize $p_{(2)}$ we compute

$$p = p(\rho_{(0)}) + p'(\rho_{(0)})\varepsilon^2 \rho_{(2)} + \mathbb{O}(\varepsilon^4) = p_{(0)} + \varepsilon^2 p_{(2)} + \mathbb{O}(\varepsilon^4).$$

After reformulation one obtains

$$p_{(2)} = p'(\rho_{(0)})\rho_{(2)} = \kappa\gamma(\rho_{(0)})^{\gamma-1}\rho_{(2)} = \frac{1}{\varepsilon^2}\kappa\gamma(\rho_{(0)})^{\gamma-1}(\rho - \rho_{(0)}). \tag{5}$$

3. RS-IMEX

In this section, we explain the basic ideas of the RS-IMEX splitting of the isentropic Euler equations for nearly incompressible flows; more details of the final algorithm are given in Section 4. Previously in Section 2, we have seen that the isentropic Euler equations (1) in the low Mach regime transform to the incompressible Euler equations (4) if $\varepsilon \to 0$. A proper numerical method should be designed in such a way that it can imitate this behavior; i.e., for $\varepsilon \to 0$ the method should formally transform into a discretization of the incompressible equations. This is the so-called *asymptotic preserving* property; see, e.g., [23].

One way to handle this type of equation is to split the flux function $\boldsymbol{F}$ into two parts and treat one part $\widehat{\boldsymbol{F}}$ with an explicit and the other part $\widetilde{\boldsymbol{F}}$ with an implicit method:

$$\partial_t \boldsymbol{w} + \nabla_x \cdot (\widetilde{\boldsymbol{F}}(\boldsymbol{w}) + \widehat{\boldsymbol{F}}(\boldsymbol{w})) = 0. \tag{6}$$

This technique results in IMEX time integration schemes; see, e.g., [2; 28] and Section 4. For stability, efficiency, and accuracy it is important to find a suitable splitting. One splitting developed in the past years is the so-called RS-IMEX, where RS stands for *reference solution*. This splitting fulfills the asymptotic preserving property in the setting of low- and high-order discretizations for the isentropic Euler equations [25; 24]. Furthermore, it gave promising results for different types of equations in the sense of stability [24; 50], efficiency [25], and accuracy [25; 24].

Definition (RS-IMEX splitting). The RS-IMEX splitting is defined by

$$\widetilde{\boldsymbol{F}}(\boldsymbol{w}) = \boldsymbol{F}(\boldsymbol{w}_{(0)}) + \boldsymbol{F}'(\boldsymbol{w}_{(0)})(\boldsymbol{w} - \boldsymbol{w}_{(0)}) \quad \text{and} \quad \widehat{\boldsymbol{F}} = \boldsymbol{F}(\boldsymbol{w}) - \widetilde{\boldsymbol{F}}(\boldsymbol{w})$$

for a given reference solution $\boldsymbol{w}_{(0)}$ and $\boldsymbol{F}'(\boldsymbol{w}_{(0)})$ the Jacobian of the flux:

$$\boldsymbol{F}'(\boldsymbol{w}_{(0)}) = \frac{\partial \boldsymbol{F}(\boldsymbol{w}_{(0)})}{\partial \boldsymbol{w}_{(0)}}.$$

In general one could choose an arbitrary reference solution, but in the following we use the limit $\boldsymbol{w}_{(0)} = \lim_{\varepsilon\to 0} \boldsymbol{w}$, which corresponds to the solution of the incompressible equation. Applying the splitting to the isentropic Euler equations gives the implicit and explicit parts

$$\widetilde{\boldsymbol{F}} = \begin{pmatrix} \rho\boldsymbol{u} \\ -\rho\boldsymbol{u}_{(0)}\otimes\boldsymbol{u}_{(0)} + \rho\boldsymbol{u}\otimes\boldsymbol{u}_{(0)} + \rho\boldsymbol{u}_{(0)}\otimes\boldsymbol{u} + \dfrac{(p(\rho_{(0)}) + p'(\rho_{(0)})(\rho-\rho_{(0)})}{\varepsilon^2}\cdot\mathbf{Id} \end{pmatrix},$$

$$\widehat{\boldsymbol{F}} = \begin{pmatrix} 0 \\ \rho(\boldsymbol{u}-\boldsymbol{u}_{(0)})\otimes(\boldsymbol{u}-\boldsymbol{u}_{(0)}) + \dfrac{p(\rho) - p(\rho_{(0)}) - p'(\rho_{(0)})(\rho-\rho_{(0)})}{\varepsilon^2}\cdot\mathbf{Id} \end{pmatrix}.$$

Considering the $\mathcal{O}(\varepsilon^{-2})$ terms, one obtains one motivation for the chosen reference solution. Since $\rho - \rho_{(0)} = \mathcal{O}(\varepsilon)$ one obtains

$$p(\rho) - p(\rho_{(0)}) - p'(\rho_{(0)})(\rho - \rho_{(0)}) = \mathcal{O}(\varepsilon^2).$$

So upon inserting the exact solution, there are no stiff terms remaining in the explicit part. Of course this is only a rationale that stands behind the scheme. Computing the eigenvalues of the nonstiff part explicitly, one obtains

$$\hat{\lambda}_{1,2} = (\boldsymbol{u} - \boldsymbol{u}_{(0)})\cdot\boldsymbol{n}, \qquad \hat{\lambda}_3 = 0, \quad \text{and} \quad \hat{\lambda}_4 = 2(\boldsymbol{u} - \boldsymbol{u}_{(0)})\cdot\boldsymbol{n},$$

and indeed, these eigenvalues are $\mathcal{O}(1)$. Even more, upon inserting the exact solution, they would be in $\mathcal{O}(\varepsilon)$ for this given choice of the reference solution. Fast waves are solely solved with the implicit method, which is a core requirement for unconditional stability with respect to ε.

Remark. A similar technique has been used for the stiff collision operator of kinetic equations in [16] and for the pressure gradient of shallow-water equations in [9; 20].

4. Discretization

4.1. ***High-order discontinuous Galerkin IMEX framework.*** Discontinuous Galerkin (DG) schemes have recently gained considerable interest as baseline schemes for multiscale problems; see, e.g., [5; 3; 11; 47] and the references therein. They can be interpreted as a hybrid finite element–finite volume formulation, where an elementwise Galerkin variational formulation is coupled weakly to its neighbors through a numerical flux term. Each inner-element solution is approximated by a polynomial function of given order $\mathcal{N}$. Penalization of discontinuities and the

locality of the basis make DG suitable for hyperbolic problems. In addition, the compact operator with small memory and communication footprint leads to excellent parallel scaling properties and the element-based approximation enables unstructured meshing of complex domains.

To obtain a DG discretization, we assume that the domain is separated into a finite number of independent cells. Then we seek a piecewise smooth function $\boldsymbol{w_h}$; i.e., it is a polynomial of maximal degree $\mathcal{N}$ on every cell, which fulfills the weak discontinuous Galerkin formulation, given by

$$\frac{\partial}{\partial t}\int_E \boldsymbol{w_h}\phi(\boldsymbol{x})\,d\boldsymbol{x} + \oint_{\partial E} \boldsymbol{F}_n^*\phi(\boldsymbol{x})\,ds - \int_E \boldsymbol{F}(\boldsymbol{w_h})\cdot\nabla_x\phi(\boldsymbol{x})\,d\boldsymbol{x} = 0, \qquad (7)$$

on every cell E for every polynomial test function $\phi(\boldsymbol{x})$ of maximal degree $\mathcal{N}$. Note that $\boldsymbol{F}_n^*$ denotes the surface normal numerical flux function, given by $\boldsymbol{F}_n^* := \boldsymbol{F}_n^*(\boldsymbol{w_h}^+, \boldsymbol{w_h}^-)$ and superscripts $\pm$ denote the values at the grid cell interface from the neighbor and the local grid cell E, respectively.

The current investigations are based on a particularly efficient variant of the general DG formulation (7), namely the discontinuous Galerkin spectral element method (DGSEM) proposed by [32]. In this formulation, the solution $\boldsymbol{w}$ is approximated as a tensor product of one-dimensional Lagrange interpolating polynomials of degree $\mathcal{N}$. The $\mathcal{N}+1$ Legendre–Gauss quadrature points $\{\xi_i\}_{i=0}^{\mathcal{N}}$ are chosen as interpolation nodes. This collocation of interpolation and integration nodes significantly reduces the number of operations per degree of freedom. In particular, the tensor product structure of the solution ansatz transfers to the operator itself, avoiding element–global volume operations. Instead, the multidimensional operator is constructed of consecutive one-dimensional operations. One disadvantage is that this reduces the flexibility of DG with respect to meshing, as only quadrilateral meshes can be used in order not to destroy the tensor product structure.

Details on the implementation and efficiency of the solver are given by Hindenlang et al. [22]. Extension of the framework to include multiphase flow based on a sharp interface approach, large eddy simulation methods, and shock capturing strategies are given by [15; 17; 7; 45]. The full FLEXI framework, including pre- and postprocessing tools, is available as open source software.[1]

For the extension of this solver to an implicit-explicit time discretization, we consider again a splitting as in (6). (Note that, with $\widetilde{\boldsymbol{F}}(\boldsymbol{w}) = \boldsymbol{F}(\boldsymbol{w})$ and $\widehat{\boldsymbol{F}}(\boldsymbol{w}) = 0$, also a fully implicit scheme falls into this framework.) IMEX schemes are defined by their Butcher tableaux featuring the coefficients $\tilde{A}$, $\hat{A}$, $\tilde{c}$, and $\hat{c}$. In semidiscrete form the implicit-explicit Runge–Kutta time discretization for the i-th stage and

[1] https://www.flexi-project.org, GNU General Public License v3.0.

the n-th time step can be written as

$$\boldsymbol{w}^{n,i} - \boldsymbol{w}^n + \Delta t \Bigg(\sum_{j=1}^{i} \tilde{A}_{i,j} \nabla_x \cdot \widetilde{\boldsymbol{F}}(\boldsymbol{w}^{n,j}, t^n + \tilde{c}_j \Delta t) + \sum_{j=1}^{i-1} \hat{\boldsymbol{A}}_{i,j} \nabla_x \cdot \widehat{\boldsymbol{F}}(\boldsymbol{w}^{n,j}, t^n + \hat{c}_j \Delta t) \Bigg) = 0. \quad (8)$$

A reformulation of (8) yields

$$\boldsymbol{w}^{n,i} + \Delta t \tilde{A}_{i,i} \nabla_x \cdot \widetilde{\boldsymbol{F}}(\boldsymbol{w}^{n,i}, t^n + \tilde{c}_i \Delta t) = \boldsymbol{w}^n - \Delta t \sum_{j=1}^{i-1} \big[\tilde{A}_{i,j} \nabla_x \cdot \widetilde{\boldsymbol{F}}(\boldsymbol{w}^{n,j}, t^n + \tilde{c}_j \Delta t) + \hat{A}_{i,j} \nabla_x \cdot \widehat{\boldsymbol{F}}(\boldsymbol{w}^{n,j}, t^n + \hat{c}_j \Delta t) \big],$$

where the right-hand side is either known from previous stages or can be computed explicitly. In the following, this equation is abbreviated by

$$(\mathbf{Id} - \Delta t \tilde{A}_{i,i} \widetilde{\boldsymbol{R}}) \boldsymbol{w}^{n,i} = \boldsymbol{b},$$

with $\widetilde{\boldsymbol{R}}$ denoting the spatial operator with the implicitly treated fluxes. To solve this potentially nonlinear (for a fully implicit scheme it is; for the RS-IMEX it is linear!) system, a standard root finding algorithm such as Newton's method can be applied. Therefore, the IMEX-Runge–Kutta scheme for the k-th Newton's iteration reads

$$\begin{aligned} \boldsymbol{w}^{(k+1)} &= \boldsymbol{w}^{(k)} + \Delta \boldsymbol{w}, \\ \Delta \boldsymbol{w} - \Delta t \tilde{A}_{i,i} \frac{\partial \widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)})}{\partial \boldsymbol{w}} \Delta \boldsymbol{w} &= -\boldsymbol{w}^{(k)} + \Delta t \tilde{A}_{i,i} \widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)}) + b. \end{aligned} \quad (9)$$

For ease of presentation, we have omitted the superscript n, i.

Equation (9) is a linear system for every Newton iteration k, which can be solved with a standard linear solving algorithm. To minimize computational costs for calculating and storing the Jacobian, the matrix-free GMRES linear solving algorithm by Saad and Schultz [40] is applied. In [18] it has been shown that a matrix-free approach can be superior to a matrix-based approach for a high-order three-dimensional DG scheme. The matrix-vector product including the Jacobian in (9) is approximated via a finite difference

$$\frac{\partial \widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)})}{\partial \boldsymbol{w}} \Delta \boldsymbol{w} \approx \frac{\widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)} + \Delta_{\mathrm{FD}} \Delta \boldsymbol{w}) - \widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)})}{\Delta_{\mathrm{FD}}} \quad (10)$$

for a small Δ_{FD} which can be calculated according to Qin et al. [37] and Knoll and Keynes [31] as

$$\Delta_{\mathrm{FD}} = \frac{\sqrt{\mathrm{eps}}}{\| \Delta \boldsymbol{w} \|_2},$$

with eps being the machine accuracy or the maximum achievable accuracy. As the implicit flux of the RS-IMEX splitting is linear, this finite difference can be simplified, but special care has to be taken of Dirichlet-type boundary conditions. Hence, for the RS-IMEX splitting, the matrix-vector product can be simplified to

$$\frac{\partial \widetilde{\boldsymbol{R}}(\boldsymbol{w}^{(k)})}{\partial \boldsymbol{w}} \Delta \boldsymbol{w} = \widetilde{\boldsymbol{R}}(\Delta \boldsymbol{w}) - \widetilde{\boldsymbol{R}}(\boldsymbol{0}). \tag{11}$$

In our implementation, we use a standard block-Jacobian preconditioner due to the small building and storing costs of the preconditioner. This turned out to be beneficial for a DG setup with a very large number of processors [8]. In [8] it is shown that more sophisticated preconditioners like full SGS and multilevel preconditioners are not superior to the standard block-Jacobian preconditioner regarding computational time for a parallel DG setup.

4.2. ***Incompressible solver.*** The RS-IMEX splitting, defined on page 247, requires the corresponding incompressible state. Therefore, an incompressible solver in the discontinuous Galerkin framework is needed. We start with the incompressible Euler equations as given in (4) in three dimensions and reformulate them as

$$\partial_t \begin{pmatrix} 0 & \\ & \mathbf{1} \end{pmatrix} \boldsymbol{U} + \nabla_x \cdot \widetilde{\boldsymbol{G}}(\boldsymbol{U}) = 0,$$

for the state vector $\boldsymbol{U} = (p_{(2)}, u_{(0),1}, u_{(0),2}, u_{(0),3})^T$ and with the flux

$$\widetilde{\boldsymbol{G}}(\boldsymbol{U}) := \begin{pmatrix} \boldsymbol{u}_{(0)} \\ \boldsymbol{u}_{(0)} \otimes \boldsymbol{u}_{(0)} + (p_{(2)}/\rho_{(0)}) \cdot \mathbf{Id} \end{pmatrix}.$$

Note as a reminder that $p_{(2)}$ denotes the hydrodynamic pressure, $\rho_{(0)}$ is a constant positive value, and $\boldsymbol{u}_{(0)} = (u_{(0),1}, u_{(0),2}, u_{(0),3})^T$ denotes the three-dimensional velocity vector. As the divergence-free condition for the velocity field is not a time evolution equation for the hydrodynamic pressure, a numerical flux function is required to couple the velocity and pressure field. A flux which satisfies this condition for solving incompressible flows with a discontinuous Galerkin scheme has been proposed by Bassi et al. [4]. In order to obtain a flux at the interfaces, artificial compressibility is added for the solution of the Riemann problem. An iterative Godunov-type Riemann solver is used in [4] to obtain the interface fluxes. Following [6] this artificial compressibility approach allows an equal-order discretization for the pressure and velocity. Moreover it is shown to be a consistent discretization of the incompressible Euler equations as the added compressibility is zero for vanishing jumps at the cell interfaces. A further advantage of this approach is that it offers the possibility to use the same high-order numerical methods for space and time discretization as for the compressible method. As the accuracy of the incompressible reference solution for the RS-IMEX splitting is not crucial, we

use a cheaper Lax–Friedrichs-type Riemann solver motivated by the asymptotic analysis, which reads

$$\widetilde{\boldsymbol{G}}^* = \frac{1}{2}\left(\widetilde{\boldsymbol{G}}(\boldsymbol{U}^+) + \widetilde{\boldsymbol{G}}(\boldsymbol{U}^-) + \mathbf{Diag}\left(\frac{\rho_{(0)}^{1-\gamma}}{\kappa\gamma}, 1, 1, 1\right)(\boldsymbol{U}^+ - \boldsymbol{U}^-)\right),$$

with κ and γ from the equation of state (2).

5. Validation and reformulation

In this section, we validate the code and give first impressions of its performance. Furthermore, we indicate how to avoid problems with machine accuracy when ε is very small. As an underlying example, we use the two-dimensional smooth traveling vortex presented in [24]. For both compressible and incompressible isentropic Euler equations, the solution is a mere transport of an initial vortex in the x_1 direction with speed 0.5. The compressible solution reads

$$\rho(\boldsymbol{x}, t) = \rho\left(\begin{pmatrix} x_1 - 0.5t \\ x_2 \end{pmatrix}, 0\right), \qquad \boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{u}\left(\begin{pmatrix} x_1 - 0.5t \\ x_2 \end{pmatrix}, 0\right),$$

for the initial conditions

$$\begin{aligned} \rho(\boldsymbol{x}, t=0) &= 2 + (500\epsilon)^2 \cdot \begin{cases} 0.5e^{2/\Delta r}\Delta r - \mathrm{Ei}(2/\Delta r) & \text{for } r < 0.5, \\ 0 & \text{otherwise}, \end{cases} \\ \boldsymbol{u}(\boldsymbol{x}, t=0) &= \begin{pmatrix} 0.5 \\ 0 \end{pmatrix} + 500 \begin{pmatrix} -x_2 + 0.5 \\ x_1 - 0.5 \end{pmatrix} \cdot \begin{cases} e^{1/\Delta r} & \text{for } r < 0.5, \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{12}$$

with $r := \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}$, $\Delta r := r^2 - 0.25$, and the equation of state $p(\rho) = \kappa\rho^\gamma$ with $\kappa = 0.5$ and $\gamma = 2$. Ei denotes the exponential integral function

$$\mathrm{Ei}(x) := \int_{-\infty}^{x} \frac{e^t}{t}\, dt.$$

In our implementation we use the algorithm by Press et al. [36] for the exponential integral function. Boundary conditions of the domain $\Omega = [0, 1]^2$ are chosen to be periodic. The initialization of the incompressible pressure is obtained via (5).

5.1. ***Validation.*** Here, we present numerical results validating the solver. As time integrators, we use the IMEX Runge–Kutta schemes IMEX-ARS-222 and IMEX-ARS-443 by [2] as second- and third-order schemes and IMEX-ARK-4A2 from [34] as a fourth-order scheme. All schemes are given with their Butcher tableaux in the Appendix; see Tables 2, 3, and 4. In the numerical results, an appropriate polynomial degree is chosen so that the overall order is the order of the time integration scheme.

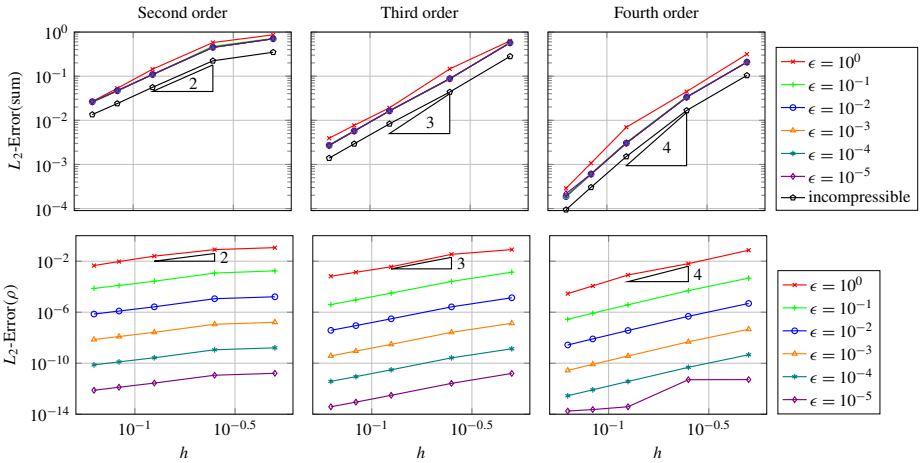


Figure 1. h-convergence of second-, third-, and fourth-order incompressible and RS-IMEX schemes for traveling vortex in overall L_2-error (top) and L_2-error in density (bottom) for different Mach numbers.

Figure 1 shows the convergence of the overall L_2-error including the errors in momentum and density (top) for the incompressible solver and the RS-IMEX splitting. Overall, the L_2 error is computed by

$$\|\boldsymbol{w_h} - \boldsymbol{w}\|^2_{L_2(\Omega)} := \int_\Omega \|\boldsymbol{w_h}(\boldsymbol{x}) - \boldsymbol{w}(\boldsymbol{x})\|_2^2 \, dx$$

where $\boldsymbol{w_h}$ is the computed numerical approximation and $\boldsymbol{w}$ the exact solution at the final time instance. Note that for the incompressible equation the error is computed in $p_{(2)}$ and $\boldsymbol{u}_{(0)}$ and for the compressible equation the error is computed in ρ and $\rho\boldsymbol{u}$. Both the incompressible solver itself and the RS-IMEX splitting which uses the incompressible solver show the correct order of convergence. Only the third-order case shows an order that is slightly too low, but this is inherent to the test case and has already been observed in [24] for under-resolved explicit calculations. Note that the overall L_2-errors for Mach numbers ranging from $\epsilon = 10^{-1}$ to $\epsilon = 10^{-5}$ nearly coincide. Additionally, Figure 1 shows the convergence in density for the RS-IMEX splitting (bottom). Here, the correct order is obtained from second to fourth order and in contrary to the overall L_2-error, the L_2-error in density scales with ε^2. This is due to the structure of the test case and the asymptotic preserving property of the method: the density can be expressed as $\rho = \text{const} + \mathbb{O}(\varepsilon^2)$, which is a disturbance in ε^2 added to a constant — this can be reproduced exactly by the DG scheme due to the AP property. Momentum can be expressed as $\rho\boldsymbol{u} = \mathbb{O}(1)$, and therefore, the error does not scale with ε.

5.2. ***Efficiency.*** In this subsection, we evaluate the efficiency of the RS-IMEX splitting in the low Mach number limit. A desirable method has the following properties.

- It is computationally cheaper than a fully implicit scheme. We have hope that this will be the case due to the linearity of the implicit flux $\widetilde{F}$.
- The scheme should — for small Mach numbers — be more efficient than a fully explicit scheme. This can also be expected, because the RS-IMEX scheme should be stable under a time step restriction that depends solely on Δx, and not on ε. An explicit scheme will always have a time step restriction of form $\Delta t \lesssim \varepsilon \Delta x$ due to the CFL condition.

For relatively large Mach numbers, we expect the RS-IMEX splitting scheme to be computationally more expensive as additional equations have to be solved. The task of this section is to identify the "sweet spot" between an explicit scheme and the RS-IMEX scheme.

We do not use the standard Lax–Friedrichs Riemann solver for the explicit and fully implicit solver as it is known to give wrong results in the low Mach number limit. The standard Lax–Friedrichs Riemann solver is defined as

$$\boldsymbol{F}^*_{\mathrm{LF}} = \tfrac{1}{2}(\boldsymbol{F}(\boldsymbol{w}^+) + \boldsymbol{F}(\boldsymbol{w}^-) + \lambda_{\max}(\boldsymbol{w}^+ - \boldsymbol{w}^-)),$$

with

$$\lambda_{\max} = \max(|\boldsymbol{u}^+ \cdot \boldsymbol{n}|, |\boldsymbol{u}^- \cdot \boldsymbol{n}|) + \frac{\max(c^+, c^-)}{\varepsilon}.$$

Inspired by the low Mach number fix for the Roe Riemann solver by Rieper [39], we utilize the low Mach Lax–Friedrichs Riemann solver

$$\boldsymbol{F}^*_{\mathrm{LF\,LMFix}} = \tfrac{1}{2}(\boldsymbol{F}(\boldsymbol{w}^+) + \boldsymbol{F}(\boldsymbol{w}^-) + \lambda_{\max}\,\mathbf{Diag}(1, \varepsilon, \varepsilon, \varepsilon)(\boldsymbol{w}^+ - \boldsymbol{w}^-)). \tag{13}$$

Note that the idea of a different scaling of density and momentum jump with respect to the Mach number has also been applied for the numerical flux of the implicit part of the RS-IMEX splitting. We show later in Section 6 that a modification multiplying the whole jump in the Riemann solver with ε is not sufficient.

We compare the computational effort for a fully implicit, a fully explicit, and the RS-IMEX scheme in Figure 2. The results have been obtained on sixteen cores with a temporal and spatial order of four. As the time integration scheme we used the IMEX-ARK-4A2 [34] for RS-IMEX, the implicit part of the same scheme for the implicit method, and a five-stage Runge–Kutta scheme [12] (see Table 6) for the explicit part. For all computations we start with the same grid and perform several refinements.

CFL numbers were chosen as $\mathrm{CFL} = 0.9$ for the explicit scheme, $\mathrm{CFL} = 150$ for the implicit scheme, and $\mathrm{CFL} = 0.5$ for the RS-IMEX scheme. For the fully

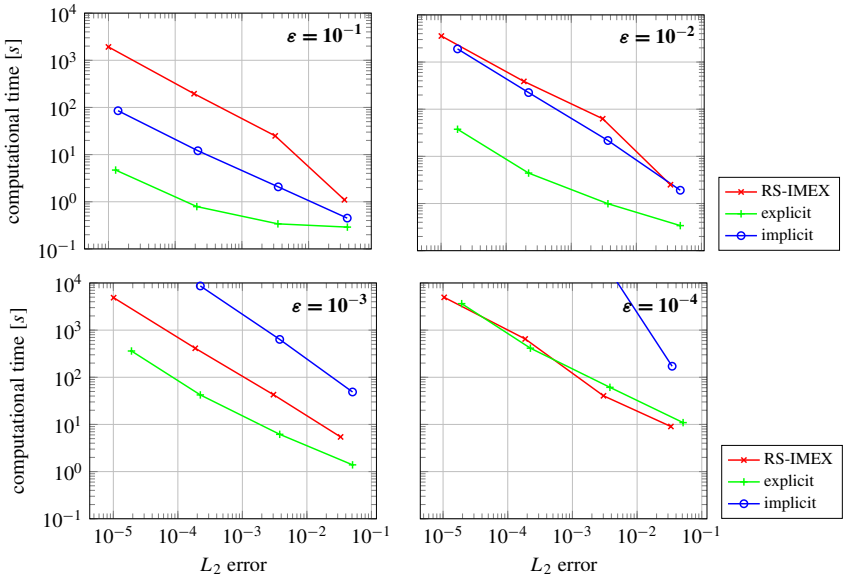


Figure 2. Comparison of computational time for two-dimensional traveling vortex (fourth order in space and time) with respect to overall L_2 error.

implicit and explicit scheme, the CFL condition is calculated using the eigenvalues of the unsplit system (3), whereas the RS-IMEX splitting only uses the convective eigenvalue ($\lambda_{1,2}$ in (3)). Our computations showed that for the implicit scheme $\mathrm{CFL} = 150$ is a good compromise between required time steps and required iterations per time step. (Note that the performance of a *linear* solver depends heavily on Δt, Δx, and ε.)

First of all, we can conclude from Figure 2 that RS-IMEX computes a smaller error on the same grid compared to the other methods (the i-th dot of each graph corresponds to the same grid).

It can be seen from Figure 2 that the computational time of the explicit and the implicit scheme scales somehow inversely to the Mach number. Since the equation system gets more and more stiff for $\varepsilon \ll 1$, the computational cost of the implicit method grows faster than the explicit ones. For the RS-IMEX only a slight increase in computational time is noticeable for a decreasing Mach number.

If the efficiency is defined as the quotient of error and computational effort, the efficiency of the explicit and implicit scheme decreases stronger than for the RS-IMEX splitting with decreasing ε due to the aforementioned scaling.

The implicit method shows an extreme growth in computational cost and therefore for $\varepsilon < 10^{-2}$ the efficiency of the implicit method becomes worse than the efficiency of the RS-IMEX method. The explicit method reaches this sweet spot for a much smaller value of ε, i.e., for $\varepsilon \leq 10^{-4}$, since the computational cost of the explicit

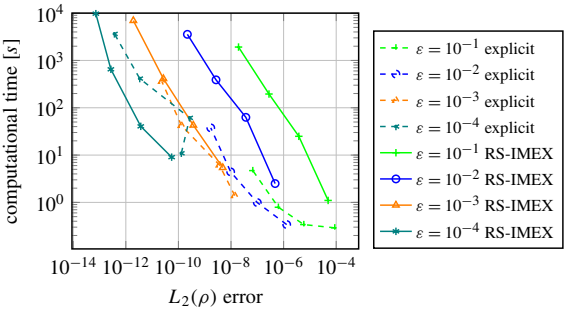


Figure 3. Comparison of computational time for 2d traveling vortex (4th order in space and time) with respect to L_2 error in density.

method is much smaller. Note that fully implicit calculations with $\varepsilon = 10^{-4}$ are very expensive as machine accuracy issues caused by the finite difference (10) lead to an extremely strong increase in computational time due to slow convergence.

We computed the same tests for a lower spatial order (second order in space and fourth order in time) and a higher spatial order (eighth order in space and fourth order in time) and obtained similar results with an earlier (low-order case) and later (higher-order case) break-even point. This behavior can be explained by the worsening of an implicit high-order scheme due to increasing storage requirements.

More improvements concerning efficiency are obtained if the error in density is considered, displayed in Figure 3. Again, the i-th symbol of each line corresponds to the same mesh. Therefore, it is visible that for low Mach numbers one obtains significantly lower errors with the RS-IMEX scheme than with the fully explicit scheme with the same mesh. The graph shows that the RS-IMEX splitting is more efficient than the explicit scheme for Mach numbers $\varepsilon < 10^{-3}$. The steepening of the $\varepsilon = 10^{-4}$ RS-IMEX line is due to round-off errors, which occur due to machine precision. We take a closer look on this problem in the next subsection.

5.3. ***Solving in the perturbation.*** It has to be noted that for very small Mach numbers, the equation becomes extremely stiff and therefore limited machine accuracy can be a problem. Indeed, in [24] the authors observed problems with the accuracy for the RS-IMEX discretization for small values of ε which cannot be explained by order reduction [10]. Similar problems have been seen in Figure 3. This observation serves as a motivation to rewrite the method similarly to the proceeding in [43]. The key trick is to rewrite the solution $\boldsymbol{w}$ as

$$\boldsymbol{w} = \underbrace{\boldsymbol{w}_{(0)}}_{\text{reference solution}} + \varepsilon \underbrace{(\boldsymbol{w}_{(1)} + \varepsilon \boldsymbol{w}_{(2)} + \mathbb{O}(\varepsilon^2))}_{\text{perturbation}} =: \boldsymbol{w}_{(0)} + \varepsilon \delta \boldsymbol{w}$$

and to observe that $\boldsymbol{w}_{(0)}$ is already part of the algorithm and therefore known, so one only has to solve in the perturbation $\delta\boldsymbol{w}$ which fulfills the equation

$$\partial_t \boldsymbol{w}_{(0)} + \varepsilon\partial_t\delta\boldsymbol{w} + \nabla_x \cdot (\widetilde{\boldsymbol{F}}(\boldsymbol{w}_{(0)} + \varepsilon\delta\boldsymbol{w}) + \widehat{\boldsymbol{F}}(\boldsymbol{w}_{(0)} + \varepsilon\delta\boldsymbol{w})) = 0.$$

In the setting of the isentropic Euler equations, $\partial_t \boldsymbol{w}_{(0)}$ can be identified by the corresponding incompressible equations. Therefore, we can replace it by the flux function $\boldsymbol{G}(\boldsymbol{w}_{(0)}, p_{(2)})$ given in (4). This results in

$$\partial_t\delta\boldsymbol{w} + \frac{1}{\varepsilon}\nabla_x \cdot (\widetilde{\boldsymbol{F}}(\boldsymbol{w}_{(0)} + \varepsilon\delta\boldsymbol{w}) - \boldsymbol{G}(\boldsymbol{w}_{(0)}, p_{(2)}) + \widehat{\boldsymbol{F}}(\boldsymbol{w}_{(0)} + \varepsilon\delta\boldsymbol{w})) = 0,$$

where $\boldsymbol{G}$ is added to the stiff part of the equation, i.e., handled with an implicit method, but does not change the implicit matrix, since the values are given. Computing the eigenvalues of the explicit part and using $\delta(\rho\boldsymbol{u}) = \rho_{(0)}\delta\boldsymbol{u} + \boldsymbol{u}\delta\rho + \varepsilon\delta\rho\delta\boldsymbol{u}$ yields

$$\hat{\lambda}_{1,2} = \varepsilon(\delta\boldsymbol{u}\cdot\boldsymbol{n}), \qquad \hat{\lambda}_3 = 0, \quad \text{and} \quad \hat{\lambda}_4 = 2\varepsilon(\delta\boldsymbol{u}\cdot\boldsymbol{n}).$$

Consequently, the explicit part has eigenvalues in $\mathbb{O}(\varepsilon)$ and the resulting method is supposed to show similar stability properties with an improved accuracy because many of the $\mathbb{O}(\varepsilon^{-1})$ terms drop out.

However, not all the terms cancel directly. One remaining term in the explicit flux is

$$\frac{1}{\varepsilon^2}(p(\rho_{(0)} + \varepsilon\delta\rho) - p(\rho_{(0)}) - p'(\rho_{(0)})\varepsilon\delta\rho).$$

Using a Taylor expansion for p gives

$$p(\rho_{(0)} + \varepsilon\delta\rho) = p(\rho_{(0)}) + \varepsilon p'(\rho_{(0)})\delta\rho + \varepsilon^2 p''(\rho_{(0)})\delta\rho^2 + \mathbb{O}(\varepsilon^3\delta\rho^3),$$

and therefore, the terms read

$$\frac{1}{\varepsilon^2}(p(\rho_{(0)}+\varepsilon\delta\rho) - p(\rho_{(0)}) - p'(\rho_{(0)})\varepsilon\delta\rho) = p''(\rho_{(0)})\delta\rho^2 + \mathbb{O}(\varepsilon\delta\rho^3) \approx p''(\rho_{(0)})\delta\rho^2.$$

We can therefore substitute the expression on the left-hand side by the one on the right-hand side; we call this proceeding *approximate pressure*. Note that — in general — this introduces an additional error in $\mathbb{O}(\varepsilon\delta\rho^3)$ to the equation, but in our setting $\delta\rho = \mathbb{O}(\varepsilon)$ and therefore the error would be in $\mathbb{O}(\varepsilon^4)$. For the low Mach case, this can safely be assumed to be negligibly small. Note furthermore that for $\gamma = 2$ this does *not* introduce an additional error.

In Figure 4 results are presented for a very small ε. Spatial and temporal accuracy is set to fourth order, i.e., we are using $\mathcal{N} = 3$ and the IMEX-ARK-4A2 scheme. We show errors for the "straightforward" RS-IMEX discretization, for solving in the perturbation only and for solving in the perturbation with an approximated pressure. Note that for the high-order vortex example the approximated pressure is an exact reformulation since $\gamma = 2$. Figure 4 shows that due to the reformulation the

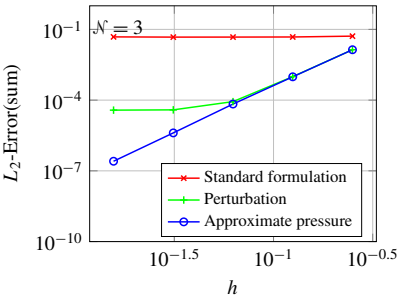


Figure 4. Convergence behavior for a fourth-order RS-IMEX discretization for the traveling vortex example for a very low Mach number of $\varepsilon = 10^{-6}$.

problems caused by machine accuracy are tremendously reduced. All computations have been done with an exact reference solution to neglect influences due to an inaccurate incompressible solver.

6. Numerical results

In this section, we present numerical results for test cases which are more physically motivated than the one considered in the previous subsection. We start with a two-dimensional flow over a cylinder, and subsequently, we investigate the three-dimensional inviscid Taylor–Green vortex.

6.1. ***Flow over a cylinder.*** This test case demonstrates the ability to use different boundary conditions in our implementation and illustrates the importance of the *asymptotic preserving* property. We compute the two-dimensional, inviscid flow over a cylinder at low Mach numbers. We apply Euler wall boundary conditions on the surface of the cylinder and Dirichlet-type boundary conditions at all other boundaries of the domain. For Euler wall boundaries we can directly prescribe the flux in normal direction at the boundaries as the normal velocity is zero:

$$\widetilde{\boldsymbol{F}}_n = \frac{(p(\rho_{(0)}) + p'(\rho_{(0)})(\rho - \rho_{(0)})}{\varepsilon^2} \begin{pmatrix} 0 \\ n_1 \\ n_2 \end{pmatrix},$$

$$\widehat{\boldsymbol{F}}_n = \frac{p(\rho) - p(\rho_{(0)}) - p'(\rho_{(0)})(\rho - \rho_{(0)})}{\varepsilon^2} \begin{pmatrix} 0 \\ n_1 \\ n_2 \end{pmatrix},$$

$$\boldsymbol{G}_n = p_{(2)} \begin{pmatrix} 0 \\ n_1 \\ n_2 \end{pmatrix},$$

whereas pressure and density are prescribed from the inner side. Dirichlet-type boundary conditions are imposed weakly, meaning that a state is prescribed on the boundaries and is used as one state required for the Riemann solver. We use a uniform two-dimensional state $\boldsymbol{w}_\infty = (\rho_\infty, u_{1,\infty}, u_{2,\infty})^T = (1.0, 1.0, 0)^T$ (in nondimensional quantities) as initialization and for the Dirichlet boundaries. For the incompressible solver the state $\boldsymbol{w}_\infty$ is transformed to $\boldsymbol{u}_{\infty,\text{incomp}} = (u_{1,\infty}, u_{2,\infty})^T = (1.0, 0)^T$, $p_{(2),\infty} = 0$, and $\rho_{(0),\infty} = \rho_\infty$. Again, the equation of state $p(\rho) = \kappa\rho^\gamma$ with $\kappa = 0.5$ and $\gamma = 2$ has been utilized. In the low Mach number limit, the exact solution is given by a potential flow field [1]. One measure of solution quality is the pressure coefficient C_p. It can be computed in two ways: once via the equation of state

$$C_p^{\text{EOS}} = \frac{1}{\varepsilon^2}\frac{p - p_\infty}{\frac{1}{2}\rho_\infty\|\boldsymbol{u}_\infty\|_2^2} = \frac{1}{\varepsilon^2}\frac{\kappa(\rho^\gamma - \rho_\infty^\gamma)}{\frac{1}{2}\rho_\infty\|\boldsymbol{u}_\infty\|_2^2}, \tag{14}$$

and once via Bernoulli's hypothesis for an incompressible, inviscid flow [1]

$$C_p^{\text{Bernoulli}} = \frac{1}{\varepsilon^2}\frac{p - p_\infty}{\frac{1}{2}\rho_\infty\|\boldsymbol{u}_\infty\|_2^2} = 1 - \frac{\rho\|\boldsymbol{u}\|_2^2}{\rho_\infty\|\boldsymbol{u}_\infty\|_2^2}. \tag{15}$$

For an incompressible, inviscid flow the result of (14) should coincide with the results of (15), and therefore should satisfy

$$C_p^{\text{EOS}} = C_p^{\text{Bernoulli}} = 1 - 4\sin^2(\theta),$$

with θ being the angular coordinate of the cylinder's polar coordinates ranging from 0 to 2π [1]. Hence, the maximum of the pressure coefficient is $C_p = 1$ at the stagnation points and the minimum $C_p = -3$ is reached at the positions with maximum velocity on the top and bottom.

Rieper [38] showed that an explicit scheme with a standard HLL-type Riemann solver reproduces the wrong pressure distribution in the low Mach number limit, as it adds too much numerical viscosity. Therefore, the explicit scheme converges to creeping flow where the dynamic pressure is several orders of magnitudes too high. In contrast, an *asymptotic preserving* scheme would reproduce the potential flow correctly. This is given since we can show for a method which is asymptotic preserving that also on the discrete level

$$\rho_h = \rho_{(0)} + \mathbb{O}(\varepsilon^2)$$

holds. Note that in this case $\rho_{(0)} = \rho_\infty$. Therefore, using a Taylor expansion in (14) we obtain

$$C_p^{\text{EOS}} = \frac{1}{\varepsilon^2}\frac{\kappa\mathbb{O}(\varepsilon^2)}{\frac{1}{2}\rho_\infty\|\boldsymbol{u}_\infty\|_2^2} = \mathbb{O}(1).$$

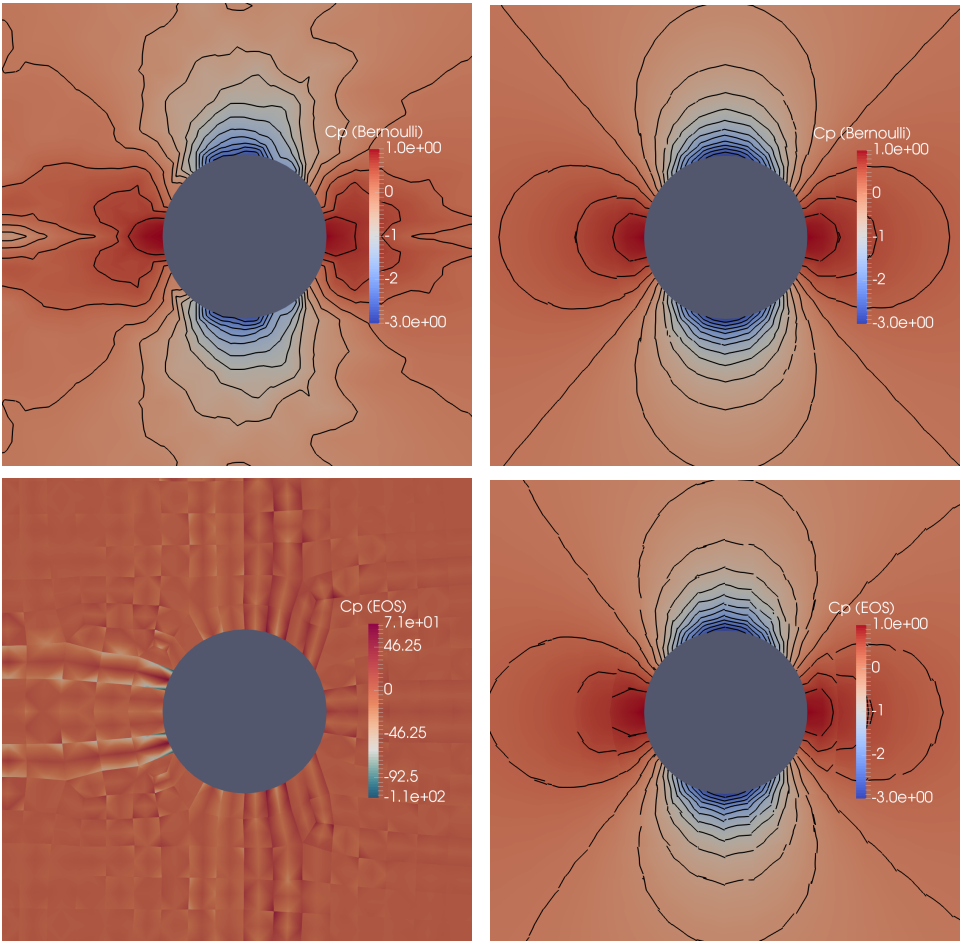


Figure 5. Isolines and colors of pressure coefficient C_p calculated via Bernoulli's hypothesis (upper) and via the equation of state (lower) for third-order explicit standard Lax–Friedrichs scheme (left) and RS-IMEX splitting (right) at $\varepsilon = 10^{-5}$.

If the method is not asymptotic preserving, the difference in the pressure might be in $\mathbb{O}(\varepsilon)$ or worse, and therefore, the pressure coefficient C_p^{EOS} becomes $\mathbb{O}(\varepsilon^{-1})$ or worse. This only affects the pressure coefficient computed via the equation of state, which is therefore an important measure of asymptotic quality of the method. Figure 5 shows the results of a calculation with 1646 elements and a polynomial degree of $\mathcal{N} = 2$ using the standard explicit Lax–Friedrichs Riemann solver on the left and the RS-IMEX splitting on the right. The Mach number is set to $\varepsilon = 10^{-5}$. If the pressure coefficient is evaluated via (15), meaning it is mainly influenced by the velocity distribution (upper row in Figure 5), both schemes are able to predict potential flow. A different behavior is observed if the dynamic pressure is evaluated

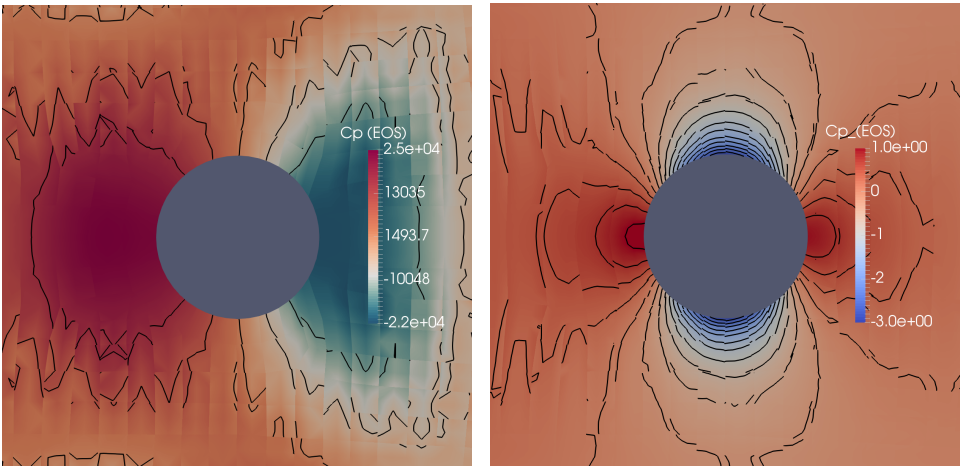


Figure 6. Isolines and colors of pressure coefficient C_p calculated via the equation of state for explicit third-order schemes with insufficient modification of the Lax–Friedrichs scheme (left) and with low Mach Lax–Friedrichs scheme (right) at $\varepsilon = 10^{-5}$.

via the equation of state (lower row). Whereas the explicit scheme with a standard Lax–Friedrichs solver does not show the correct flow pattern and has a pressure coefficient several orders or magnitude too high, the RS-IMEX scheme is able to reproduce the potential flow. This illustrates the *asymptotic preserving* property of the scheme. We use the low Mach fix proposed in (13) to show its similar asymptotic behavior compared to the asymptotic preserving RS-IMEX scheme. Figure 6 illustrates that the simple multiplication of the jump with ε is not sufficient (left) as it shows the flow pattern of a creeping flow. However, the explicit scheme with the low Mach Lax–Friedrichs Riemann solver (right) is able to predict potential flow. A further validation of the RS-IMEX splitting can be seen in Figure 7 where the C_p distribution on the upper surface of the cylinder evaluated with the equation of state and with Bernoulli's hypothesis is compared with the solution for potential flow.

6.2. ***Taylor–Green vortex.*** The Taylor–Green vortex introduced in [46] is originally a three-dimensional, incompressible viscous test case to study the transition to turbulence and its decay. For nonviscous equation systems such as the isentropic Euler equations it can be used to investigate the amount of dissipation added by a numerical scheme. The standard incompressible initial conditions are given by

$$\begin{aligned} \rho_{(0)} &= 1, \\ \boldsymbol{u}_{(0)}(\boldsymbol{x}, t=0) &= V_0 \begin{pmatrix} \cos(x_1)\cos(x_2)\cos(x_3) \\ -\cos(x_1)\sin(x_2)\cos(x_3) \\ 0 \end{pmatrix}, \\ p_{(2)}(\boldsymbol{x}, t=0) &= \frac{\rho_{(0)} V_0^2}{16}(\cos(2x_1)+\cos(2x_2))(\cos(2x_3)+2), \end{aligned}$$

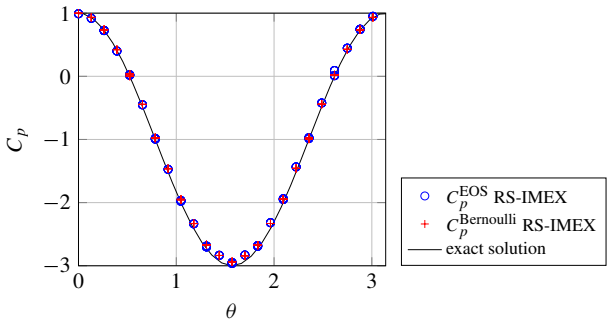


Figure 7. Distribution of C_p^{EOS} and $C_p^{\text{Bernoulli}}$ for third-order RS-IMEX splitting at $\varepsilon = 10^{-5}$ calculated with the equation of state or Bernoulli's equation in comparison with potential flow.

where V_0 denotes a constant initial velocity which is chosen to be $V_0 = 1$; x_1, x_2, and x_3 denote the spatial coordinates of the periodic box $\Omega = [0, 2\pi]^3$. We adapt the initialization for the compressible isentropic Euler equations according to (5) to obtain a consistent initial dataset for the incompressible initialization

$$\rho(\boldsymbol{x}, t = 0) = \rho_{(0)} + \varepsilon^2 \frac{V_0^2 \rho_{(0)}^{2-\gamma}}{16\gamma\kappa}(\cos(2x_1) + \cos(2x_2))(\cos(2x_3) + 2),$$

$$\boldsymbol{u}(\boldsymbol{x}, t = 0) = V_0 \begin{pmatrix} \cos(x_1)\cos(x_2)\cos(x_3) \\ -\cos(x_1)\sin(x_2)\cos(x_3) \\ 0 \end{pmatrix},$$

with $p = \kappa\rho^\gamma$, $\kappa = 0.5$, and $\gamma = 2$. All calculations were conducted on a regular grid with 16^3 elements and a polynomial degree of $\mathcal{N} = 3$. For the temporal discretization, the third-order IMEX-ARS-443 scheme by Ascher et al. [2] is used. Again, a fully implicit method is obtained if only the implicit Butcher tableau is considered. The explicit calculations were made with a standard three-stage third-order Runge–Kutta scheme [48] (see Table 5). We consider the isosurfaces of the velocity field to compare the results of the RS-IMEX splitting with the explicit scheme in a qualitative manner. Figure 8 exemplarily shows the velocity field at a Mach number of $\varepsilon = 10^{-4}$ for two different times t. In the top row, the solutions of both the explicit and the RS-IMEX scheme are identical. For consistent schemes, this is to be expected, since at this early (pretransition) state, the chosen discretization is sufficient to completely resolve the occurring scales. This notion is also supported in Figure 9, where the kinetic energy, defined as

$$E_{\text{kin,comp}} = \frac{\varepsilon^2}{2} \int_\Omega \rho \|\boldsymbol{u}\|_2^2 \, d\Omega,$$

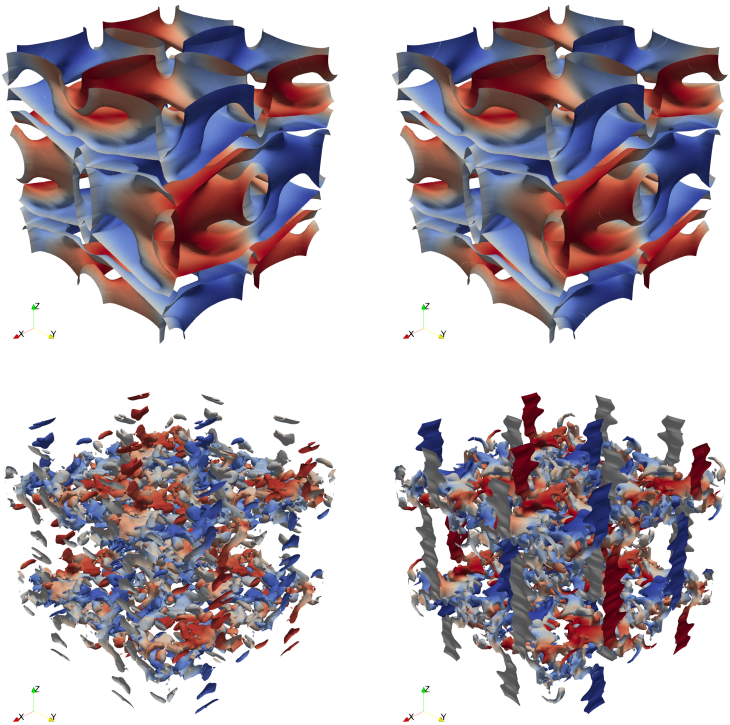


Figure 8. Isosurfaces of velocity magnitude at a physical time of $t = 3$ (top) and $t = 7$ (bottom) for explicit (left) and RS-IMEX scheme (right) at $\varepsilon = 10^{-4}$.

is preserved at $t = 3$ for both schemes. The kinetic energy can be used as a benchmark of numerical dissipation properties of a scheme for inviscid flows. In the bottom row of Figure 8, the solutions for $t = 7$ are shown. Here, clear qualitative differences exist and the kinetic energy is no longer conserved, which can be attributed to the different numerical dissipation mechanisms at work in both schemes. Calculations with other Mach numbers showed analogous results and can be seen as a further validation of the RS-IMEX scheme.

Comparisons of the dissipation rate with the compressible kinetic energy as a measure of quality, displayed in Figure 9, confirm that the explicit scheme with low-Mach Riemann solver and the RS-IMEX method behave similarly in this setting. Differences are due to the slightly different numerical dissipation added by the Riemann solvers. It is visible that a non-asymptotic preserving scheme as the explicit scheme with standard Lax–Friedrichs Riemann solver shows a Mach number dependent behavior which is not desirable. Concluding, we see that the RS-IMEX splitting is able to reproduce a complex three-dimensional physical behavior such as the Taylor–Green vortex.

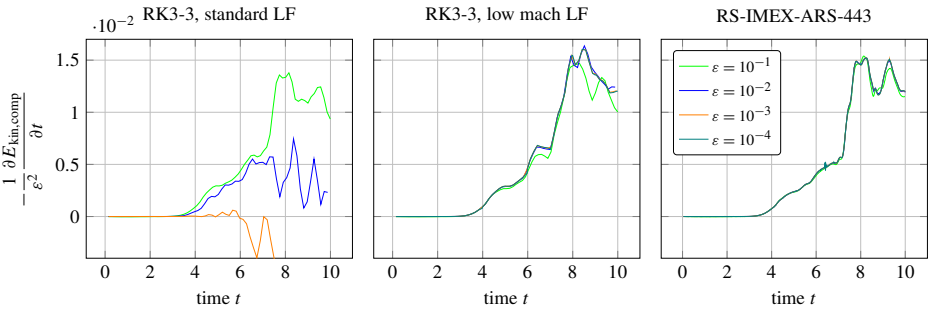


Figure 9. Scaled change rate of compressible kinetic Energy for TGV with explicit scheme with standard Lax–Friedrichs Riemann solver (left), explicit scheme with low Mach Lax–Friedrichs Riemann solver (middle), and RS-IMEX (right) (all fourth-order in space).

Δt_{init}	explicit	implicit	RS-IMEX
$\varepsilon = 10^{-1}$	$1.47 \cdot 10^{-3}$	$2.68 \cdot 10^{-1}$	$3.87 \cdot 10^{-2}$
$\varepsilon = 10^{-2}$	$1.53 \cdot 10^{-4}$	$2.79 \cdot 10^{-2}$	$3.87 \cdot 10^{-2}$
$\varepsilon = 10^{-3}$	$1.53 \cdot 10^{-5}$	$2.80 \cdot 10^{-3}$	$3.87 \cdot 10^{-2}$
$\varepsilon = 10^{-4}$	$1.53 \cdot 10^{-6}$		$3.87 \cdot 10^{-2}$

Table 1. Initial time steps of calculations with explicit, implicit, and RS-IMEX scheme for the TGV at different Mach numbers and $\Delta x = \pi/8$.

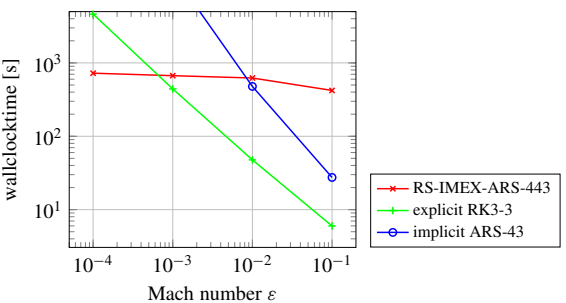


Figure 10. Comparison of computational time with 528 cores of RS-IMEX splitting and explicit and implicit schemes for TGV with 16^3 spatial elements and fourth-order in space.

Focusing on the question of efficiency, the required time for calculations with different discretization methods for several Mach numbers is displayed in Figure 10. The corresponding time steps summarized in Table 1 are given by a constant CFL number for each scheme. Computational effort increases with decreasing Mach number for the explicit scheme as the time step decreases accordingly. A strong increase of computational effort for the fully implicit scheme is noticeable as the

stiffness of the equation system increases. For "high" Mach numbers, more computational time is needed for the RS-IMEX scheme as an additional partial differential equation has to be approximated. But only a slight increase in computational effort for decreasing Mach number is observed as the stiffness is hidden in the linear system instead of the nonlinear system as for a fully implicit discretization. This constitutes obviously a huge advantage of the RS-IMEX splitting compared to a fully implicit scheme. Whereas the stiffness of the fully implicit scheme is increased in the nonlinear system, the Jacobian-vector product in (9) has to be approximated via the finite difference (10). The approximation of the Jacobian-vector product with the finite difference gets worse for an increasing stiffness of the equation system, and therefore, computational time strongly increases for the fully implicit scheme. Using the RS-IMEX splitting the Jacobian-vector product can be calculated exactly with (11). Hence, an increasing stiffness only slightly increases the computational effort. Consequently, large savings concerning computational costs can be obtained by using the RS-IMEX splitting for very low Mach numbers $\varepsilon < 10^{-3}$ compared to the explicit scheme and $\varepsilon < 10^{-2}$ compared to the implicit scheme.

7. Conclusion and outlook

The efficient and accurate numerical solution of physical phenomena that belong to the class of singularly perturbed problems is still an area of active research. These problems can be seen as a special case of multiscale problems, in which large differences in scale with regards to the average state occur in a spatially confined region of the solution. This becomes especially challenging when high accuracy in the limit is sought, i.e., the discretization should obey the underlying asymptotic properties of the equation.

In this work, we have taken steps towards the development of an efficient high-order DG scheme for all-speed flows at an engineering scale. Starting from the novel operator splitting technique RS-IMEX for the isentropic Euler equations proposed in [25], we have reformulated the discrete equations to significantly extend the Mach number range of the scheme without the occurrence of machine accuracy problems and demonstrated its capability to prevent a stall in convergence.

The RS-IMEX splitting has been implemented in an existing high-order DGSEM framework. The incompressible reference solution is solved by an artificial compressibility–type scheme, which couples the velocity and pressure field through a numerical flux function and thereby introduces a hyperbolic equation for the pressure. Numerical results have shown the efficiency of the method also in the context of realistic three-dimensional applications.

Since the RS-IMEX is conceptually independent from the underlying equations, its naive application to other systems is straightforward. However, it is not a priori

0	0	0	0	0	0	0	0
γ	0	γ	0	γ	γ	0	0
1	0	$1-\gamma$	γ	1	δ	$1-\delta$	0
	0	$1-\gamma$	γ		δ	$1-\delta$	0

Table 2. Second-order scheme IMEX-ARS-222 [2] with $\gamma = (2-\sqrt{2})/2 \approx 0.293$ and $\delta = 1 - 1/(2\gamma) \approx -0.707$.

0	0	0	0	0	0	0	0	0	0	0	0
1/2	0	1/2	0	0	0	1/2	1/2	0	0	0	0
2/3	0	1/6	1/2	0	0	2/3	11/18	1/18	0	0	0
1/2	0	−1/2	1/2	1/2	0	1/2	5/6	−5/6	1/2	0	0
1	0	3/2	−3/2	1/2	1/2	1	1/4	7/4	3/4	−7/4	0
	0	3/2	−3/2	1/2	1/2		1/4	7/4	3/4	−7/4	0

Table 3. Third-order scheme IMEX-ARS-443 [2].

clear whether this splitting guarantees hyperbolicity of the explicit part. Current research efforts are underway to answer this question and to explore the possibilities of extending the splitting to the full Euler equations. Furthermore, the application of the splitting to multiphase flows is of current interest.

Acknowledgments

The authors would like to thank Sebastian Noelle and Nico Krais for the fruitful discussions.

Jonas Zeifang has been supported by the German Research Foundation (DFG) through the International Research Training Group GRK 2160: Droplet Interaction Technologies (DROPIT). The computations with the FLEXI framework have been conducted on the Cray XC40 at the High Performance Computing Center Stuttgart under the *hpcdg* project.

Klaus Kaiser has been partially supported by the German Research Foundation (DFG) through project NO 361/6-1; his study was supported by the Special Research Fund (BOF) of Hasselt University.

Appendix

For the purpose of completeness, we list the Runge–Kutta schemes we have used throughout this paper; see Tables 2, 3, 4, 5, and 6. The left tableaux of the IMEX-Runge–Kutta schemes denote the Butcher tableaux of the part treated implicitly $\widetilde{(\cdot)}$; the right Butcher tableaux correspond to the explicit part $\widehat{(\cdot)}$.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1/3	−1/6	1/2	0	0	0	0	0	1/3	1/3	0	0	0	0	0	0
1/3	1/6	−1/3	1/2	0	0	0	0	1/3	1/6	1/6	0	0	0	0	0
1/2	3/8	−3/8	0	1/2	0	0	0	1/2	1/8	0	3/8	0	0	0	0
1/2	1/8	0	3/8	−1/2	1/2	0	0	1/2	1/8	0	3/8	0	0	0	0
1	−1/2	0	3	−3	1	1/2	0	1	1/2	0	−3/2	0	2	0	0
1	1/6	0	0	0	2/3	−1/2	2/3	1	1/6	0	0	0	2/3	1/6	0
	1/6	0	0	0	2/3	−1/2	2/3		1/6	0	0	0	2/3	1/6	0

Table 4. Fourth-order scheme IMEX-ARK-4A2 [34].

i	A_i	B_i	c_i
1	0	1/3	0
2	−5/9	15/16	1/3
3	−153/128	8/15	3/4

Table 5. Third-order low-storage explicit Runge–Kutta scheme [48].

i	A_i	B_i	c_i
1	0	$\frac{1432997174477}{9575080441755}$	0
2	$-\frac{567301805773}{1357537059087}$	$\frac{5161836677717}{13612068292357}$	$\frac{1432997174477}{9575080441755}$
3	$-\frac{2404267990393}{2016746695238}$	$\frac{1720146321549}{2090206949498}$	$\frac{2526269341429}{6820363962896}$
4	$-\frac{3550918686646}{2091501179385}$	$\frac{3134564353537}{4481467310338}$	$\frac{2006345519317}{3224310063776}$
5	$-\frac{1275806237668}{842570457699}$	$\frac{2277821191437}{14882151754819}$	$\frac{2802321613138}{2924317926251}$

Table 6. Fourth-order low-storage explicit Runge–Kutta scheme [12]

The explicit schemes are given in the $2N$-storage form [12] for the coefficients A_i, B_i, and c_i.

References

[1] J. D. Anderson, Jr., *Fundamentals of aerodynamics*, 3rd ed., McGraw-Hill, 2001.

[2] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri, *Implicit-explicit Runge–Kutta methods for time-dependent partial differential equations*, Appl. Numer. Math. **25** (1997), no. 2–3, 151–167. MR Zbl

[3] F. Bassi, L. Botti, A. Colombo, D. A. Di Pietro, and P. Tesini, *On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations*, J. Comput. Phys. **231** (2012), no. 1, 45–65. MR Zbl

[4] F. Bassi, A. Crivellini, D. A. Di Pietro, and S. Rebay, *An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations*, J. Comput. Phys. **218** (2006), no. 2, 794–815. MR Zbl

[5] F. Bassi and S. Rebay, *A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations*, J. Comput. Phys. **131** (1997), no. 2, 267–279. MR Zbl

[6] F. Bassi, A. Crivellini, D. A. Di Pietro, and S. Rebay, *An implicit high-order discontinuous Galerkin method for steady and unsteady incompressible flows*, Comput. & Fluids **36** (2007), no. 10, 1529–1546. MR Zbl

[7] A. D. Beck, T. Bolemann, D. Flad, H. Frank, G. J. Gassner, F. Hindenlang, and C.-D. Munz, *High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations*, Internat. J. Numer. Methods Fluids **76** (2014), no. 8, 522–548. MR

[8] P. Birken, G. Gassner, M. Haas, and C.-D. Munz, *Preconditioning for modal discontinuous Galerkin methods for unsteady 3D Navier–Stokes equations*, J. Comput. Phys. **240** (2013), 20–35. MR Zbl

[9] G. Bispen, K. R. Arun, M. Lukáčová-Medvid'ová, and S. Noelle, *IMEX large time step finite volume methods for low Froude number shallow water flows*, Commun. Comput. Phys. **16** (2014), no. 2, 307–347. MR Zbl

[10] S. Boscarino, *Error analysis of IMEX Runge–Kutta methods derived from differential-algebraic systems*, SIAM J. Numer. Anal. **45** (2007), no. 4, 1600–1621. MR Zbl

[11] A. Buffa, T. J. R. Hughes, and G. Sangalli, *Analysis of a multiscale discontinuous Galerkin method for convection-diffusion problems*, SIAM J. Numer. Anal. **44** (2006), no. 4, 1420–1440. MR Zbl

[12] M. H. Carpenter and C. A. Kennedy, *Fourth-order 2N-storage Runge–Kutta schemes*, technical memorandum 109112, National Aeronautics and Space Administration, 1994.

[13] P. Degond and M. Tang, *All speed scheme for the low Mach number limit of the isentropic Euler equations*, Commun. Comput. Phys. **10** (2011), no. 1, 1–31. MR Zbl

[14] D. R. Durran, *A physically motivated approach for filtering acoustic waves from the equations governing compressible stratified flow*, J. Fluid Mech. **601** (2008), 365–379. MR Zbl

[15] S. Fechter and C.-D. Munz, *A discontinuous Galerkin-based sharp-interface method to simulate three-dimensional compressible two-phase flow*, Internat. J. Numer. Methods Fluids **78** (2015), no. 7, 413–435. MR

[16] F. Filbet and S. Jin, *A class of asymptotic-preserving schemes for kinetic equations and related problems with stiff sources*, J. Comput. Phys. **229** (2010), no. 20, 7625–7648. MR Zbl

[17] D. Flad, A. Beck, and C.-D. Munz, *Simulation of underresolved turbulent flows by adaptive filtering using the high order discontinuous Galerkin spectral element method*, J. Comput. Phys. **313** (2016), 1–12. MR Zbl

[18] M. Franciolini, A. Crivellini, and A. Nigro, *On the efficiency of a matrix-free linearly implicit time integration strategy for high-order discontinuous Galerkin solutions of incompressible turbulent flows*, Comput. & Fluids **159** (2017), 276–294. MR

[19] F. X. Giraldo and M. Restelli, *High-order semi-implicit time-integrators for a triangular discontinuous Galerkin oceanic shallow water model*, Internat. J. Numer. Methods Fluids **63** (2010), no. 9, 1077–1102. MR Zbl

[20] F. X. Giraldo, M. Restelli, and M. Läuter, *Semi-implicit formulations of the Navier–Stokes equations: application to nonhydrostatic atmospheric modeling*, SIAM J. Sci. Comput. **32** (2010), no. 6, 3394–3425. MR Zbl

[21] J. Haack, S. Jin, and J.-G. Liu, *An all-speed asymptotic-preserving method for the isentropic Euler and Navier–Stokes equations*, Commun. Comput. Phys. **12** (2012), no. 4, 955–980. MR Zbl

[22] F. Hindenlang, G. J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, and C.-D. Munz, *Explicit discontinuous Galerkin methods for unsteady problems*, Comput. & Fluids **61** (2012), 86–93. MR Zbl

[23] S. Jin, *Asymptotic preserving (AP) schemes for multiscale kinetic and hyperbolic equations: a review*, Riv. Math. Univ. Parma (N.S.) **3** (2012), no. 2, 177–216. MR Zbl

[24] K. Kaiser and J. Schütz, *A high-order method for weakly compressible flows*, Commun. Comput. Phys. **22** (2017), no. 4, 1150–1174. MR

[25] K. Kaiser, J. Schütz, R. Schöbel, and S. Noelle, *A new stable splitting for the isentropic Euler equations*, J. Sci. Comput. **70** (2017), no. 3, 1390–1407. MR Zbl

[26] S. Kawai, *Direct numerical simulation of transcritical turbulent boundary layers at supercritical pressures with strong real fluid effects*, 54th AIAA Aerospace Sciences Meeting, American Institute of Aeronautics and Astronautics, 2016, 2016-1934.

[27] S. Kawai, H. Terashima, and H. Negishi, *A robust and accurate numerical method for transcritical turbulent flows at supercritical pressure with an arbitrary equation of state*, J. Comput. Phys. **300** (2015), 116–135. MR Zbl

[28] C. A. Kennedy and M. H. Carpenter, *Additive Runge–Kutta schemes for convection-diffusion-reaction equations*, Appl. Numer. Math. **44** (2003), no. 1–2, 139–181. MR Zbl

[29] S. Klainerman and A. Majda, *Singular limits of quasilinear hyperbolic systems with large parameters and the incompressible limit of compressible fluids*, Comm. Pure Appl. Math. **34** (1981), no. 4, 481–524. MR Zbl

[30] R. Klein, *Semi-implicit extension of a Godunov-type scheme based on low Mach number asymptotics, I: One-dimensional flow*, J. Comput. Phys. **121** (1995), no. 2, 213–237. MR Zbl

[31] D. A. Knoll and D. E. Keyes, *Jacobian-free Newton–Krylov methods: a survey of approaches and applications*, J. Comput. Phys. **193** (2004), no. 2, 357–397. MR Zbl

[32] D. A. Kopriva, *Implementing spectral methods for partial differential equations: algorithms for scientists and engineers*, Springer, 2009. MR Zbl

[33] S.-H. Lee, *Cancellation problem of preconditioning method at low Mach numbers*, J. Comput. Phys. **225** (2007), no. 2, 1199–1210. Zbl

[34] H. Liu and J. Zou, *Some new additive Runge–Kutta methods and their applications*, J. Comput. Appl. Math. **190** (2006), no. 1–2, 74–98. MR Zbl

[35] H. Paillere, C. Viozat, A. Kumbaro, and I. Toumi, *Comparison of low Mach number models for natural convection problems*, Heat Mass Transfer **36** (2000), no. 6, 567–573.

[36] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in Fortran 77: the art of scientific computing*, 2nd ed., Cambridge University, 1996.

[37] N. Qin, D. K. Ludlow, and S. T. Shaw, *A matrix-free preconditioned Newton/GMRES method for unsteady Navier–Stokes solution*, Int. J. Numer. Meth. Fl. **33** (2000), no. 2, 223–248. Zbl

[38] F. Rieper, *On the dissipation mechanism of upwind-schemes in the low Mach number regime: a comparison between Roe and HLL*, J. Comput. Phys. **229** (2010), no. 2, 221–232. MR Zbl

[39] F. Rieper, *A low-Mach number fix for Roe's approximate Riemann solver*, J. Comput. Phys. **230** (2011), no. 13, 5263–5287. MR Zbl

[40] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput. **7** (1986), no. 3, 856–869. MR Zbl

[41] S. Schochet, *The mathematical theory of low Mach number flows*, M2AN Math. Model. Numer. Anal. **39** (2005), no. 3, 441–458. MR Zbl

[42] J. Schütz and K. Kaiser, *A new stable splitting for singularly perturbed ODEs*, Appl. Numer. Math. **107** (2016), 18–33. MR Zbl

[43] J. Sesterhenn, B. Müller, and H. Thomann, *On the cancellation problem in calculating compressible low Mach number flows*, J. Comput. Phys. **151** (1999), no. 2, 597–615. MR Zbl

[44] J. R. Simões Moreira and J. E. Shepherd, *Evaporation waves in superheated dodecane*, J. Fluid Mech. **382** (1999), 63–86.

[45] M. Sonntag and C.-D. Munz, *Efficient parallelization of a shock capturing for discontinuous Galerkin methods using finite volume sub-cells*, J. Sci. Comput. **70** (2017), no. 3, 1262–1289. MR Zbl

[46] G. I. Taylor and A. E. Green, *Mechanism of the production of small eddies from large ones*, P. Roy. Soc. Lond. A Mat. **158** (1937), no. 895, 499–521. JFM

[47] Z. J. Wang, K. Fidkowski, R. Abgrall, and et al., *High-order CFD methods: current status and perspective*, Internat. J. Numer. Methods Fluids **72** (2013), no. 8, 811–845. MR

[48] J. H. Williamson, *Low-storage Runge–Kutta schemes*, J. Comput. Phys. **35** (1980), no. 1, 48–56. MR Zbl

[49] W.-A. Yong, *A note on the zero Mach number limit of compressible Euler equations*, Proc. Amer. Math. Soc. **133** (2005), no. 10, 3079–3085. MR Zbl

[50] H. Zakerzadeh and S. Noelle, *A note on the stability of implicit-explicit flux-splittings for stiff systems of hyperbolic conservation laws*, Commun. Math. Sci. **16** (2018), no. 1, 1–15. Zbl

Received May 24, 2017. Revised March 28, 2018.

JONAS ZEIFANG: zeifang@iag.uni-stuttgart.de
Institut für Aerodynamik und Gasdynamik, Universität Stuttgart, Stuttgart, Germany

KLAUS KAISER: kaiser@igpm.rwth-aachen.de
Institut für Geometrie und Praktische Mathematik, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen, Germany

and

Faculteit Wetenschappen, Universiteit Hasselt, Diepenbeek, Belgium

ANDREA BECK: beck@iag.uni-stuttgart.de
Institut für Aerodynamik und Gasdynamik, Universität Stuttgart, Stuttgart, Germany

JOCHEN SCHÜTZ: jochen.schuetz@uhasselt.be
Faculteit Wetenschappen, Universiteit Hasselt, Diepenbeek, Belgium

CLAUS-DIETER MUNZ: munz@iag.uni-stuttgart.de
Institut für Aerodynamik und Gasdynamik, Universität Stuttgart, Stuttgart, Germany

Communications in Applied Mathematics and Computational Science

msp.org/camcos

See inside back cover or msp.org/camcos for submission instructions.

The subscription price for 2018 is US $100/year for the electronic version, and $150/year (+$15, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues from the last three years and changes of subscriber address should be sent to MSP.

Communications in Applied Mathematics and Computational Science (ISSN 2157-5452 electronic, 1559-3940 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

CAMCoS peer review and production are managed by EditFLOW® from MSP.

PUBLISHED BY

mathematical sciences publishers

nonprofit scientific publishing

http://msp.org/

Communications in Applied Mathematics and Computational Science

vol. 13 no. 2 2018