Communications on Applied Mathematics and Computation (2025) 7:3–39
https://doi.org/10.1007/s42967-023-00264-y

ORIGINAL PAPER

On Error-Based Step Size Control for Discontinuous Galerkin Methods for Compressible Fluid Dynamics

Hendrik Ranocha[1] · Andrew R. Winters[2] · Hugo Guillermo Castro[3] · Lisandro Dalcin[3] · Michael Schlottke-Lakemper[4,5] · Gregor J. Gassner[6,7] · Matteo Parsani[3,8]

Received: 15 September 2022 / Revised: 2 February 2023 / Accepted: 15 February 2023 /
Published online: 22 May 2023

Abstract
We study a temporal step size control of explicit Runge-Kutta (RK) methods for compressible computational fluid dynamics (CFD), including the Navier-Stokes equations and hyperbolic systems of conservation laws such as the Euler equations. We demonstrate that error-based approaches are convenient in a wide range of applications and compare them to more classical step size control based on a Courant-Friedrichs-Lewy (CFL) number. Our numerical examples show that the error-based step size control is easy to use, robust, and efficient, e.g., for (initial) transient periods, complex geometries, nonlinear shock capturing approaches, and schemes that use nonlinear entropy projections. We demonstrate these properties for problems ranging from well-understood academic test cases to industrially relevant large-scale computations with two disjoint code bases, the open source Julia packages Trixi.jl with OrdinaryDiffEq.jl and the C/Fortran code SSDC based on PETSc.

Keywords Explicit Runge-Kutta (RK) methods · Step size control · Compressible fluid dynamics · Adaptivity in space and time · Shock capturing

Mathematics Subject Classification 65L06 · 65M20 · 65M70 · 76M10 · 76M22 · 76N99 · 35L50

1 Introduction

Hyperbolic conservation laws are at the heart of many physical models in science and engineering, e.g., fluid dynamics describing the airflow around an airplane, acoustics, and space plasma modeling. Usually, the systems need to be solved numerically. A common approach is to apply the method of lines, i.e., to apply a spatial semi-discretization first and solve the resulting ordinary differential equation (ODE) numerically. For this task, explicit

✉ Hendrik Ranocha
mail@ranocha.de

Extended author information available on the last page of the article

Runge-Kutta (RK) methods are among the most commonly used schemes because of their efficiency and parallel scalability [48, 53, 55].

Since demanding simulations require a significant amount of compute resources, there is a strong interest in developing efficient numerical methods. Here, the efficiency can be measured by the required compute time, energy, and human time. The time- and energy-to-solution depend on the algorithms and their implementation as well as the hardware [4, 41, 93] and are often a focus of high performance computing (HPC) studies. The required human time depends on the robustness of the overall numerical method and the sensitivity with respect to parameters that may need to be tuned to ensure the robustness and/or the efficiency. In this article, we study an error-based step size control for compressible flow problems to demonstrate that it is efficient in all three aspects.

We use entropy-dissipative semi-discretizations to ensure the robustness [29, 31, 83, 86] but do not investigate specific implementation techniques discussed elsewhere [33, 61, 77]. Instead, we focus on the step size control of time integration methods applied to ODEs resulting from spatial semi-discretizations of compressible flow problems. The goal is to adapt the time step size Δt such that it is as big as possible while still satisfying the accuracy and stability requirements. For explicit RK methods applied to (high-order) semi-discretizations, the stability requirements are usually more restrictive. Thus, a commonly used approach is to adapt the time step size according to a Courant-Friedrichs-Lewy (CFL) number chosen by the user. This involves estimating local wave speeds and mesh spacing, which can be demanding for complex systems and curved high-order meshes in multiple space dimensions required in practice. The optimal CFL number depends on the space and time discretizations and possibly on the specific problem [14, 58, 84]. An alternative, widely used in the context of general-purpose ODE solvers, is an error-based step size control. There, a local error estimate is obtained from an embedded method and fed into a controller adapting the time step size according to user-specified tolerances. This approach has also been used successfully for partial differential equations [6, 51, 74, 94] but appears to be less-widely applied in the CFD community.

In this article, we systematically study an error-based step size control for compressible CFD problems and compare it to CFL-based approaches. Specifically, we use the RK methods and controllers developed in [74] in two different software environments. Most of the examples use the open source packages OrdinaryDiffEq.jl [73] for ODE solvers and Trixi.jl [78, 85] for spatial semi-discretizations implemented in Julia [7]. Some large-scale, industrially relevant CFD simulations are implemented in SSDC [67], which is built on top of the Portable and Extensible Toolkit for Scientific computing (PETSc) [3], its mesh topology abstraction (DMPLEX) [51], and its scalable ODE/DAE solver library [1]. All source code and instructions required to reproduce the numerical experiments using open source packages are available online in our reproducibility repository [79].

In the following, we briefly review RK methods and step size control techniques in Sect. 2. Afterwards, we study the robustness and the sensitivity to user-supplied parameters under a change of mesh structure of both step size control approaches in Sect. 3. Section 4 focuses on the effect of nonlinear shock capturing schemes. In Sect. 5, we discuss the impact of an initial transient period, e.g., in the cold start of a simulation initialized with free stream values. Thereafter, we study the step size control in the presence of a change of variables in Sect. 6. Next, we discuss the convenience of the error-based step size control for exploratory research of new systems in Sect. 7. Finally, we summarize and discuss our findings in Sect. 8.

2 Runge-Kutta (RK) Methods and Step Size Control

We use the method of lines approach and first discretize a conservation law in space. In particular, we focus on spatial semi-discretizations using collocated discontinuous Galerkin (DG) methods (and related shock capturing schemes); see, e.g., [29, 31, 83, 86]. The resulting ODE

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t}u(t) = f(t, u(t)), \quad t \in (0, T), \\ u(0) = u^0 \end{cases} \tag{1}$$

is then solved using numerical time integration methods. Here, we use explicit RK methods with an embedded error estimator of the form [12, 38]

$$\begin{cases} y_i &= u^n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} f(t_n + c_j \Delta t_n, y_j), \qquad i \in \{1, \cdots, s\}, \\ u^{n+1} &= u^n + \Delta t_n \sum_{i=1}^{s} b_i f(t_n + c_i \Delta t_n, y_i), \\ \widehat{u}^{n+1} &= u^n + \Delta t_n \sum_{i=1}^{s} \widehat{b}_i f(t_n + c_i \Delta t_n, y_i) + \widehat{b}_{s+1} f(t_{n+1}, u^{n+1}). \end{cases} \tag{2}$$

Here, u^{n+1} is the numerical solution of the main method, $\widehat{u}^{n+1}$ is the embedded methods solution used to obtain the local error estimate $u^{n+1} - \widehat{u}^{n+1}$, and y_i are the stage values. RK methods are parameterized by their Butcher tableau

$$\begin{array}{c|c} c & A \\ \hline & b^{\mathrm{T}} \\ & \widehat{b}^{\mathrm{T}} \end{array} \tag{3}$$

where $A = (a_{ij})_{i,j=1}^{s} \in \mathbb{R}^{s \times s}$ is strictly lower triangular for explicit methods, $b, c \in \mathbb{R}^s$, and $\widehat{b} \in \mathbb{R}^{s+1}$. If $\widehat{b}_{s+1} \neq 0$, the RK pair (2) uses the so-called first-same-as-last (FSAL) idea to include the predicted value at the new time in the embedded error estimator [27]. This can increase the performance of the embedded method and comes at no additional cost if the step is accepted, since $f(t_{n+1}, u^{n+1})$ must be computed as the first stage of the next step anyway.

All methods considered in this article are applied in the local extrapolation mode, i.e., a main method of order q is coupled with an embedded error estimator of order $\widehat{q} = q - 1$. Whenever possible, we make use of low-storage formats such as 3S*$_+$ [50, 74]. We use the same naming convention of RK methods as in [74], i.e., NAME$q(\widehat{q})s$ indicates an s-stage method of order q with the embedded error estimator of order $\widehat{q}$. This base name is followed by additional information on low-storage properties and a subscript “F” for FSAL methods.

2.1 Time Integration Loop and Callbacks

We use implementations of the time integration methods available in OrdinaryDiffEq.jl [73] in Julia or PETSc/TS [1]. The basic steps for the time integration loop are given in Algorithm 1. At first, a preliminary time step is performed with the given RK method. If the error-based step size control is activated, the embedded error estimator is also computed; it is used to update the time step size and determine whether the time step is rejected. After accepting a time step, callbacks are activated. We use these callback mechanisms for adaptive mesh refinement (AMR) and, if activated, CFL-based step size control.

Algorithm 1 High-level overview of a time integration step in the Julia package OrdinaryDiffEq.jl

```
i:  Compute preliminary time step update $u^{n+1}$ and the embedded solution
    $\widehat{u}^{n+1}$ if the error-based step size control is activated using (2).
ii: If the error-based step size control is activated:
    • let the step size controller update the time step size;
    • if the time step is rejected, repeat step i with the updated time step size.
iii: Accept the new time step and update internal caches.
iv:  Activate callbacks.
v:   Proceed to the next time step (i.e., increment the time step number$n$ and
     go to step i).
```

A more detailed overview of the time integration loop is given below in Algorithm 2, discussed after introducing the step size control mechanisms.

2.2 CFL-Based Step Size Control

Explicit time integration methods for first-order hyperbolic conservation laws are subject to a CFL time step restriction of the form $\Delta t \leqslant C\Delta x/\lambda_{\max}$ [22]. However, it is non-trivial to provide sharp estimates of the terms in this restriction, e.g., an appropriate value for Δx on curved meshes for high-order DG methods in multiple space dimensions. In general, the CFL-based step size control requires user estimates of the maximal local wave speeds $\lambda_{\max}(u_i^n)$ at the degree of freedom u_i^n and the corresponding local mesh spacing Δx_i. Then, the time step is determined as

$$\Delta t_n = \nu \min_i \frac{\Delta x_i}{\lambda_{\max}(u_i^n)}, \tag{4}$$

where ν is the desired CFL number. For DG methods using polynomials of degree p on (possibly) curved grids for the linear advection equation $\partial_t u + (a \cdot \partial_x)u = 0$, we follow [2, 74] and use the estimate

$$\frac{\Delta x_i}{\lambda_{\max}(u_i^n)} = \frac{2}{p+1} \frac{J_i}{\sum_{j=1}^{d} |(J\partial_x \xi^j)_i \cdot a|}, \tag{5}$$

where d is the number of spatial dimensions, J_i the determinant of the grid Jacobian $\partial_x \xi$ at node i, $(J\partial_x \xi^j)_i$ the contravariant basis vector in direction j at node i [54, Chapter 6], and 2 is the size of the reference element. For nonlinear conservation laws, a is replaced by appropriate estimates of the local wave speeds.

2.3 Error-Based Step Size Control

We perform the error-based step size control based on controller ideas from digital signal processing [36, 37, 88–90]. In particular, we use PID controllers that select a new time step based on

$$\Delta t_{n+1} = \kappa\left(\epsilon_{n+1}^{\beta_1/k} \epsilon_n^{\beta_2/k} \epsilon_{n-1}^{\beta_3/k}\right) \Delta t_n, \tag{6}$$

where $k = \min(q, \widehat{q}) + 1$, q is the order of the main method, and $\widehat{q}$ is the order of the embedded method. Because $\widehat{q} = q - 1$ in our case, we have that $k = \widehat{q} + 1 = q$. Moreover, β_i are the controller parameters, κ is a step size limiter, and

$$\epsilon_{n+1} = \frac{1}{w_{n+1}}, \qquad w_{n+1} = \left(\frac{1}{m} \sum_{i=1}^{m} \left(\frac{u_i^{n+1} - \widehat{u}_i^{n+1}}{\tau_a + \tau_r \max\{|u_i^{n+1}|, |\widetilde{u}_i|\}} \right)^2 \right)^{1/2}, \tag{7}$$

where m is the total number of degrees of freedom in u, and τ_a, τ_r are the absolute and relative error tolerances. In OrdinaryDiffEq.jl, $\widetilde{u} = u^n$, while PETSc uses $\widetilde{u} = \widehat{u}^{n+1}$. Unless stated otherwise, we use equal tolerances $\tau_a = \tau_r = \texttt{tol}$ and the default step size limiter $\kappa(a) = 1 + \arctan(a - 1)$ [91].

The decision whether the step shall be accepted or rejected is determined by the size of the factor multiplying the time step size Δt_n in (6). The default option used for all numerical experiments is to accept a step whenever the limited factor is at least 0.9^2. Otherwise, the step is rejected and the time step size Δt_n is set as the new predicted step size (6). Another common strategy is to decide whether a step should be rejected based only on the current local error estimate. Söderlind and Wang [89, 91] argued why the approach we described here can be beneficial to reduce the amount of (unnecessary but expensive) step rejections.

To make the step size control fully automatic, we use the estimate of the initial step size implemented in OrdinaryDiffEq.jl from the algorithm described by Hairer et al. [38, p. 169]. ϵ_n and ϵ_{n-1} are initially set to one (for $n = 0$).

A more detailed overview of the time integration loop including additional aspects about the step size control and step rejections is shown in Algorithm 2. Parameters such as the threshold 0.9^2 used to determine whether a step should be accepted or the limiting threshold $w_{\min}$ used in Algorithm 2 are based on recommendations in reference works such as [38, Sect. IV.2] and [12, Sect. 371]; we always use their default values and do not consider them as user-facing parameters that need to be chosen manually.

Algorithm 2 High-level overview of a time integration loop using explicit RK methods with embedded error estimator and PID controllers in OrdinaryDiffEq.jl and deviations in PETSc/TS

```
Initialize time $t \leftarrow 0$, time step number $n \leftarrow 0$, and initial state $u^0$.
Initialize PID controller with $\varepsilon_0 \leftarrow 1$, $\varepsilon_{-1} \leftarrow 1$.
Initialize $\Delta t_0 = \widetilde{\Delta t}$ with a given value or the algorithm of [38, p. 169].
Initialize accept_step $\leftarrow$ false
while $t < T$ do
    if accept_step then                      ▷ callbacks can change accept_step
        accept_step $\leftarrow$ false               ▷ prepare for the next step
        $t \leftarrow \widetilde{t}$
        $\Delta t_{n+1} \leftarrow \widetilde{\Delta t}$
        $n \leftarrow n + 1$
    else
        $\Delta t_n \leftarrow \widetilde{\Delta t}$
    end if
    if $t + \Delta t_n > T$ then
        $\Delta t_n \leftarrow T - t$
    end if
    Compute $u^{n+1}$ and $\widehat{u}^{n+1}$ with time step $\Delta t_n$
    $w_{n+1} \leftarrow \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\frac{u_i^{n+1}-\widehat{u}_i^{n+1}}{\tau_a+\tau_r \max\left\{\left|u_i^{n+1}\right|,\left|\widetilde{u}_i\right|\right\}}\right)^2}$
                        ▷ $\widetilde{u} = u^n$ in OrdinaryDiffEq.jl, $\widetilde{u} = \widehat{u}^{n+1}$ in PETSc
    $w_{n+1} \leftarrow \max\{w_{n+1}, w_{\min}\}$
      ▷ $w_{\min} \approx 2.2 \times 10^{-16}$ in OrdinaryDiffEq.jl, $= 1.0 \times 10^{-10}$ in PETSc
    $\varepsilon_{n+1} \leftarrow 1/w_{n+1}$
    dt_factor $\leftarrow \kappa\left(\varepsilon_{n+1}^{\beta_1/k}\varepsilon_n^{\beta_2/k}\varepsilon_{n-1}^{\beta_3/k}\right)$   ▷ default $\kappa(a) = 1 + \arctan(a - 1)$
    $\widetilde{\Delta t} \leftarrow$ dt_factor $\cdot \Delta t_n$
    if dt_factor $\geqslant$ accept_safety then ▷ default accept_safety $= 0.81$
        accept_step $\leftarrow$ true
    else
        accept_step $\leftarrow$ false
    end if
    if accept_step then
        $\widetilde{t} \leftarrow t + \Delta t_n$
        if $\widetilde{t} \approx T$ then ▷ within 100 units in last place in OrdinaryDiffEq.jl
            $\widetilde{t} \leftarrow T$
        end if
        Apply callbacks   ▷ AMR, CFL-based control in OrdinaryDiffEq.jl
    end if
end while
```

2.4 Importance of Controller Parameters

The RK pair (2) and the PID controller (6) must be developed together to obtain good results. In particular, semi-discretizations of conservation laws limited by stability should be integrated using a combination of RK pair and controller leading to a stable step size control, based on the test equation [40]; see also [39, Sect. IV.2] and [49]. This has been demonstrated for spectral element discretizations for compressible flows in [74]. In this article, we use the optimized controllers of [74].

2.5 Representative Runge-Kutta (RK) Methods Used in This Article

There is a vast amount of literature on RK methods. Many schemes have been designed as general-purpose methods for low- or high-tolerance applications or even specifically for hyperbolic conservation laws. While we have performed numerical experiments with various schemes, we restrict this article to the following representative set of methods. First, we consider the general-purpose method:

- BS3(2)$_\mathrm{F}$, the third-order, three-stage method of [8] (`BS3()` in OrdinaryDiffEq.jl, `3bs` in PETSc).

This method has been shown to be among the most efficient—if not the most efficient—general-purpose method for the CFD problems in which we are interested [74]. Next, we consider methods optimized for spectral element semi-discretizations of compressible CFD, namely,

- RDPK3(2)5$_\mathrm{F}$[3S$^*{}_+$], the third-order, five-stage method of [74] (`RDPK3SpFSAL35()` in OrdinaryDiffEq.jl);
- RDPK4(3)9$_\mathrm{F}$[3S$^*{}_+$], the fourth-order, nine-stage method of [74] (`RDPK3SpFSAL49()` in OrdinaryDiffEq.jl).

Finally, we consider the strong stability preserving (SSP) method:

- SSP3(2)4[3S$^*{}_+$], the third-order, four-stage SSP method of [57] with embedded method of [21] and efficient implementation and step size controller of [74] (`SSPRK43()` in OrdinaryDiffEq.jl).

3 Robustness Under Change of Mesh Structure

Both CFL- and error-based step size controls come with parameters that must be chosen by the user, either a CFL factor ν or a tolerance `tol`. However, the sensitivity with respect to these parameters differs significantly. The CFL factor ν influences the time step sizes—and thus the efficiency—linearly, while the tolerance `tol` has roughly a logarithmic influence. Thus, it is arguably easier to choose an appropriate tolerance than an optimal CFL factor.

Moreover, in the stability limited regime, it is often convenient to use the error-based step size control, since there is usually a range of tolerances resulting in the same number of right-hand side (RHS) evaluations a manually optimized CFL-control could achieve at best. Furthermore, an optimal CFL factor ν depends on the mesh. Thus, it can vary when introducing curved coordinates compared to a uniform grid. This was already demonstrated for a 2D linear advection equation in [74, Sect. 3]. Here, we extend this demonstration to the nonlinear compressible Euler equations of an ideal gas in $d = 3$ space dimensions given by

$$\partial_t \begin{pmatrix} \rho \\ \rho v_i \\ \rho e \end{pmatrix} + \sum_{j=1}^{d} \partial_{x_j} \begin{pmatrix} \rho v_j \\ \rho v_i v_j + p\delta_{ij} \\ (\rho e + p) v_j \end{pmatrix} = 0, \tag{8}$$

where ρ is the density, v the velocity, ρe the total energy, and p the pressure given by

$$p = (\gamma - 1)\Big(\rho e - \frac{1}{2}\rho v^2\Big), \tag{9}$$

with the ratio of specific heats $\gamma = 1.4$. Specifically, we consider the classical inviscid Taylor-Green vortex with the initial data[1]

$$\begin{cases} \rho^0 = 1,\ v_1^0 = \sin(x_1)\cos(x_2)\cos(x_3),\ v_2^0 = -\cos(x_1)\sin(x_2)\cos(x_3),\ v_3^0 = 0, \\ p^0 = \frac{\rho^0}{Ma^2\gamma} + \rho^0 \frac{\cos(2x_1)\cos(2x_3)+2\cos(2x_2)+2\cos(2x_1)+\cos(2x_2)\cos(2x_3)}{16}, \end{cases} \tag{10}$$

where $Ma = 0.1$ is the Mach number. We consider the domain $[-\pi, \pi]^3$ with periodic boundary conditions and meshes with 8 elements per coordinate direction. We apply entropy-dissipative DG methods with polynomials of degree $p = 3$ using a local Lax-Friedrichs flux at interfaces and the flux of [75, 80, 81] in the volume terms. We integrate the semi-discretizations in the time interval [0, 10].

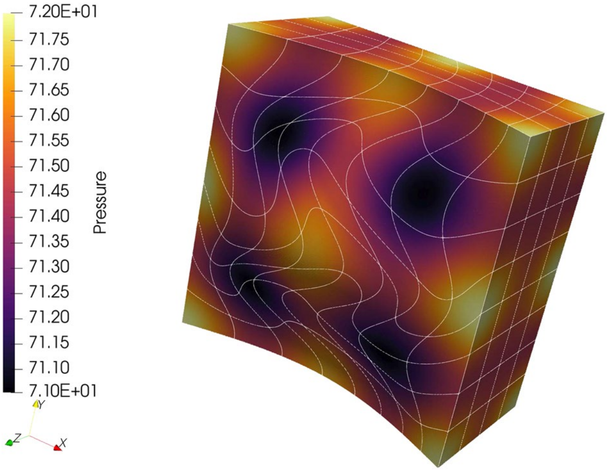
We consider two meshes, a uniform Cartesian mesh and a curved mesh that heavily warps the Cartesian reference coordinates $(\xi, \eta, \zeta) \in [-1, 1]^3$ to the desired mesh in physical coordinates (x, y, z) with the mapping

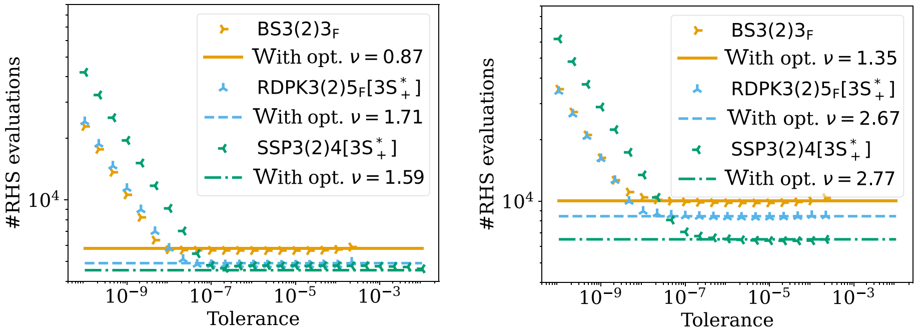
$$\begin{cases} y = \eta + \dfrac{L_y}{8}\big(\cos(3\pi(\xi - c_x)/L_x)\cos(\pi(\eta - c_y)/L_y)\cos(\pi(\zeta - c_z)/L_z)\big), \\ x = \xi + \dfrac{L_x}{8}\big(\cos(\pi(\xi - c_x)/L_x)\cos(4\pi(y - c_y)/L_y)\cos(\pi(\zeta - c_z)/L_z)\big), \\ z = \zeta + \dfrac{L_y}{8}\big(\cos(\pi(x - c_x)/L_x)\cos(2\pi(y - c_y)/L_y)\cos(\pi(\zeta - c_z)/L_z)\big), \\ L_x = L_y = L_z = 2\pi, \quad c_x = c_y = c_z = 0, \end{cases} \tag{11}$$

which has been adapted from [17]. The curved mesh and the initial pressure are visualized in Fig. 1.

[1] We use a superscript 0 to denote the initial data.

Fig. 1 Initial pressure of the compressible Euler equations for the inviscid Taylor-Green vortex on a slice of the curved mesh





(a) Cartesian mesh

(b) Curved mesh

Fig. 2 Number of ODE RHS evaluations for representative RK methods using error-based (markers) and CFL-based (lines) step size control. The CFL factors were tuned manually to be as large as possible without crashing the simulation. Note that the optimized CFL factors are between 55% and 74% larger on the curved grid

The required number of RHS evaluations for the selected, representative RK methods is shown in Fig. 2, for CFL-based and error-based step size controls. All RK methods show the same qualitative behavior on both grids. The step size for this problem is largely limited by the stability, resulting in a range of tolerances yielding the optimal number of RHS evaluations one can also achieve by tuning the CFL factor manually (maximizing ν to three significant digits so that the simulations do not crash). This range of tolerances spans up to five orders of magnitude. For looser tolerances, the simulations may crash due to positivity issues, typically in the first few time step. Stricter tolerances let the controller detect an accuracy-limited regime and increase the number of RHS evaluations accordingly. This happens at looser tolerances for SSP3(2)4[3S$^*{}_+$] than for the other methods, since SSP3(2)4[3S$^*{}_+$] is only optimized for the SSP coefficient, while the other methods are also constructed to minimize the principal truncation error. Since all methods are of the same order of accuracy, the slope of the increasing number of RHS evaluations is the same.

However, there is a significant difference between CFL-based and error-based step size controls when switching the mesh types. Indeed, the CFL factor can be increased between 55% (for BS3(2)3_{F}) and 74% (for SSP3(2)4[$3\mathrm{S}^{*}{}_{+}$]) on the curved mesh compared to the equivalent Cartesian mesh. In particular, choosing an optimized CFL factor from the uniform mesh results in the same amount of efficiency loss when the CFL factor is not tuned again manually from scratch. In contrast, basically, the same range of tolerances can be used to get the optimal number of RHS evaluations on both grids for the error-based step size control.

4 Change of Linear Stability Restrictions in Nonlinear Schemes

There are many shock capturing approaches for DG methods, e.g., artificial viscosity [35, 72], replacing DG elements by finite volume subcells [28, 87], modal filtering [63, 76], or specially constructed invariant domain preserving methods [34, 59, 71]. Here, we use the a priori convex blending of high-order DG elements with finite volume subcells described in [42].

4.1 Linear CFL Restrictions

Shock capturing approaches in nonlinear schemes will typically change the linear CFL restriction on the time step for explicit time integration schemes. Here, we demonstrate this for some third-order schemes of the representative RK methods introduced in Sect. 2.5.

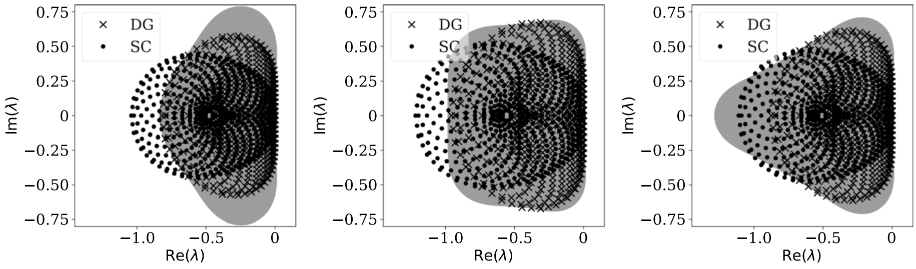
We consider the linear advection equation

$$\partial_t u(t,x) + \sum_{j=1}^{2} a^j \partial_j u(t,x) = 0 \tag{12}$$

in the domain $[-1,1]^2$ with periodic boundary conditions and velocity $a = (1,1)^{\mathrm{T}}/\sqrt{2}$. We discretize the domain using 8^2 uniform elements with polynomials of degree $p = 3$ on Legendre-Gauss-Lobatto nodes. We choose a fixed blending parameter $\alpha = 0.5$ in the shock capturing scheme of [42] with the standard DG collocation spectral element method (DGSEM), e.g., [54], and local Lax-Friedrichs flux for the finite volume subcells. We compute the spectra of the standard DGSEM scheme and the shock capturing scheme. Due to the fixed choice of the blending parameter, both semi-discretizations are linear and we compute their spectra numerically.

The spectra embedded into the stability regions of the representative RK methods are visualized in Fig. 3. To make it easier to compare the RK methods, the stability regions are scaled by the effective number of RK stages, taking the FSAL property into account. The spectra of the semi-discretizations are scaled by the maximal factor, so that the standard DGSEM spectrum is in the region of absolute stability of the RK method. Clearly, the (scaled) spectra of the shock capturing semi-discretizations are partially outside of the stability regions. Thus, one can expect a more restrictive CFL condition than for the standard DGSEM schemes.

These linear CFL restrictions predict an approximately one-quarter bigger CFL number for standard DGSEM discretizations compared to the shock capturing variants for BS3(2)3_{F} and RDPK3(2)5_{F}[$3\,\mathrm{S}^{*}{}_{+}$]. For these time integration methods, the SC spectra extend significantly to the left-half of the complex plane outside of the stability regions, in



(a) BS3(2)3_{F} (b) RDPK3(2)5_{F}[3S*$_+$] (c) SSP3(2)4[3S*$_+$]

Fig. 3 Spectra of DGSEM and shock capturing (SC) semi-discretizations of the linear advection equation are embedded into the stability regions of some representative RK methods. The stability regions are scaled by the effective number of stages of the RK methods (taking the FSAL property into account). The spectra of the semi-discretizations are scaled by the maximal factor, so that the standard DGSEM spectrum is in the region of absolute stability

particular near the negative real axis. The effect is qualitatively similar but quantitatively less pronounced for SSP3(2)4[3S*$_+$] (ca. 5% instead of more than 20%).

4.2 Application to Nonlinear Magnetohydrodynamics: Orszag-Tang Vortex

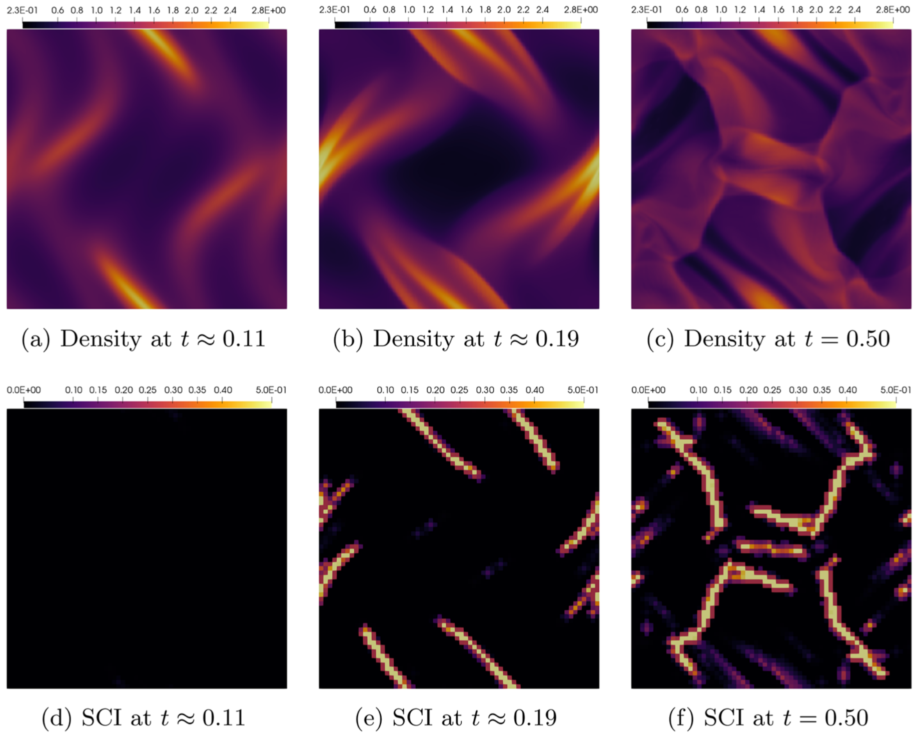
Next, we apply the representative time integration methods to entropy-dissipative semi-discretizations of the ideal generalized Lagrange multiplier (GLM) magnetohydrodynamics (MHD) equations [9, 25] for the Orszag-Tang vortex setup [66]. Specifically, we use the initial condition

$$\begin{cases} \rho^0 = 1, \quad v_1^0 = -\sin(2\pi x_2), \quad v_2^0 = \sin(2\pi x_1), \quad v_3^0 = 0, \quad p^0 = 1/\gamma, \\ B_1^0 = -\sin(2\pi x_2)/\gamma, \quad B_2^0 = \sin(4\pi x_1)/\gamma, \quad B_3^0 = 0, \quad \psi^0 = 0, \end{cases} \tag{13}$$

in the domain $[0, 1]^2$ with periodic boundary conditions, where ρ is the density, v the velocity, p the pressure, B the magnetic field, and ψ the Lagrange multiplier to control divergence errors. The ideal GLM-MHD equations with ratio of specific heats $\gamma = 5/3$ and divergence cleaning speed $c_h = 1$ are discretized on a uniform mesh with $2^6 = 64$ elements per coordinate direction and polynomials of degree $p = 3$. We use the shock capturing method of [42] with the entropy-conservative numerical flux of [44] and a local Lax-Friedrichs flux at interfaces and for finite volume subcells, both with the nonconservative Powell source term. We use the product of density and pressure as the shock indicator variable with blending parameters $\alpha_{\max} = 0.5$ and $\alpha_{\min} = 0.001$.

As shown in Fig. 4, the flow and its numerical approximation remain smooth in its initial phase. Between $t = 0.1$ and $t = 0.2$, the shock capturing indicator detects troubled cells and activates the finite volume shock capturing mechanism. At the final time, several shocklets are visible in the approximation.

We recorded the time step sizes and effective CFL numbers, i.e., the CFL factor computed a posteriori based on the time step size chosen by the error-based step size controller, after every accepted time step for some representative RK methods. The tolerance is set to $\texttt{tol} = 10^{-4}$ in all cases with the error-based step size control. For the CFL-based step size control, we bisected the maximal CFL number without blow-up



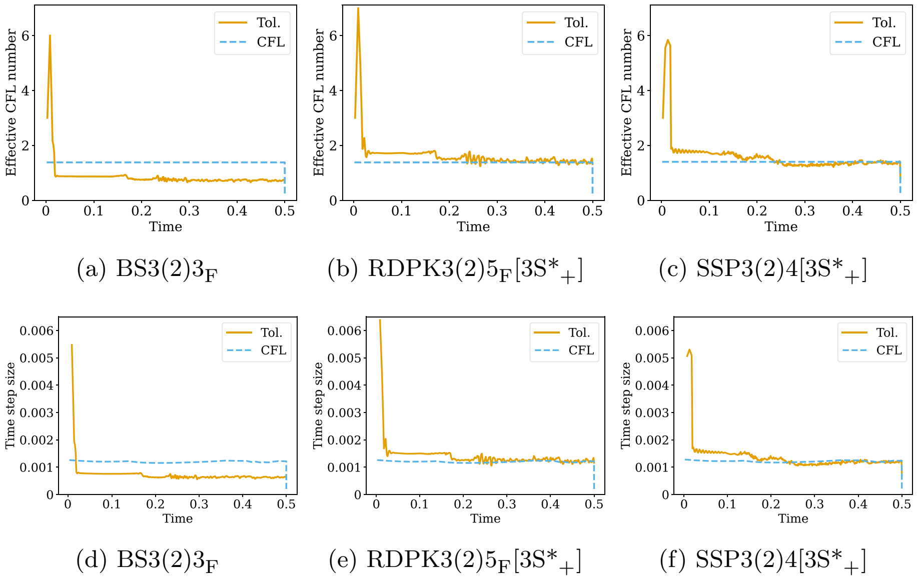
(a) Density at $t \approx 0.11$ (b) Density at $t \approx 0.19$ (c) Density at $t = 0.50$

(d) SCI at $t \approx 0.11$ (e) SCI at $t \approx 0.19$ (f) SCI at $t = 0.50$

Fig. 4 Evolution of the density and the shock capturing indicator (SCI) computed using SSP3(2)4[3S$^*{}_+$] and entropy-dissipative shock capturing semi-discretizations of the ideal GLM-MHD equations for the Orszag-Tang vortex

to three significant digits. The results are shown in Fig. 5. Clearly, the time step size of the CFL-based approaches does not vary significantly in time. Thus, the estimated CFL restriction is nearly constant. However, the nonlinear schemes activate shock capturing mechanisms between $t = 0.1$ and $t = 0.2$. Thus, the estimated linear CFL restriction should become more severe as discussed in Sect. 4.1. This can be observed also for the error-based step size control; all RK methods use larger time steps initially until the shock capturing part is activated. Then, the error-based step size control yields approximately the maximal step size also used by the CFL-based step size control with manually tuned CFL number. In particular, the initial time step size is approximately one-quarter bigger with the error-based step size control (after the controller has adapted the time step size from the automated initial guess). During the full simulation, the error-based step size control results in the following performance improvements based on the number of function evaluations (see also Table 1): 13% for BS3(2)3$_\text{F}$, 11% for RDPK3(2)5$_\text{F}$[3S$^*{}_+$], and 6% for SSP3(2)4[3S$^*{}_+$].

While the quantitative numbers for BS3(2)3$_\text{F}$ and RDPK3(2)5$_\text{F}$[3S$^*{}_+$] fit very well to the linear analysis, one could expect that the initial advantage of SSP3(2)4[3S$^*{}_+$] with the error-based step size control should be smaller, since the stability region is less optimal for high-order DGSEM and relatively better suited for low-order shock capturing methods. First, we do not necessarily expect that such a linear analysis is quantitatively



(a) BS3(2)3_F (b) RDPK3(2)5_F[3S$^*_+$] (c) SSP3(2)4[3S$^*_+$]

(d) BS3(2)3_F (e) RDPK3(2)5_F[3S$^*_+$] (f) SSP3(2)4[3S$^*_+$]

Fig. 5 Effective CFL numbers (first row) and time step sizes (second row) of some representative RK methods applied to entropy-dissipative shock capturing semi-discretizations of the ideal GLM-MHD equations for the Orszag-Tang vortex

correct for general nonlinear problems, although it was shown to behave very well in applications [70]. Second, the shock capturing mechanism also changes the behavior of the ODE RHS, usually reducing the smoothness of the spatially discrete system, which also affects step size control.

As expected, the method RDPK3(2)5_F[3S$^*_+$] optimized for spectral element discretizations performs better than the general-purpose method BS3(2)3_F and uses 15% less function evaluations (with the error-based step size control). For this problem with a significant amount of shock capturing, SSP3(2)4[3S$^*_+$] with error-based step size performs even better and uses additionally 16% less function evaluations than RDPK3(2)5_F[3S$^*_+$].

In summary, the results of this section demonstrate that the error-based step size control can react robustly and efficiently to varying linear CFL restrictions in nonlinear

Table 1 Performance of representative RK methods with default error-based and manually tuned CFL-based step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for the Orszag-Tang vortex with entropy-dissipative shock capturing semi-discretizations

Scheme	β	`tol`/ν	#FE	#A	#R
BS3(2)3_F	(0.60, – 0.20, 0.00)	`tol` = 10^{-4}	2 187	724	4
		$\nu = 0.69$	2 506	835	
RDPK3(2)5_F[3S$^*_+$]	(0.70, – 0.23, 0.00)	`tol` = 10^{-4}	1 863	368	4
		$\nu = 1.38$	2 091	418	
SSP3(2)4[3S$^*_+$]	(0.55, – 0.27, 0.05)	`tol` = 10^{-4}	1 550	384	3
		$\nu = 1.40$	1 648	412	

schemes. In particular, they usually require less parameter tuning while still achieving optimal performance. For this problem with a significant amount of shock capturing mechanisms, the strong stability preserving method SSP3(2)4[3S$^*{}_+$] performs well.

5 Initial Transient Period: Cold Start of a Simulation

When performing a cold start of a demanding CFD simulation, e.g., by initializing the flow around objects with free stream values, the initial transient period often requires smaller time steps than the fully developed simulation. For demanding simulations with positivity issues for high-order spatial schemes used in this section, we apply SSP time integration methods.

5.1 Double Mach Reflection of a Strong Shock

First, we consider the double Mach reflection of Woodward and Colella [97]. We set up an initial grid with 80 uniform elements that is adaptively refined during the simulation using `p4est` [11]. We use the shock capturing approach of [42] with DG elements using polynomials of degree $p = 4$ and the entropy-conservative numerical flux of [75, 80, 81] in the volume terms and a local Lax-Friedrichs flux at interfaces and finite volume subcells. AMR is triggered every two time steps and the positivity-preserving limiter of [99] for density and pressure is applied after every RK stage. We apply SSP3(2)4[3S$^*{}_+$] to integrate the system in time $t \in [0, 0.2]$. The complete setup can be found in the reproducibility repository [79]. Figure 6 shows the density and the mesh at the final time.

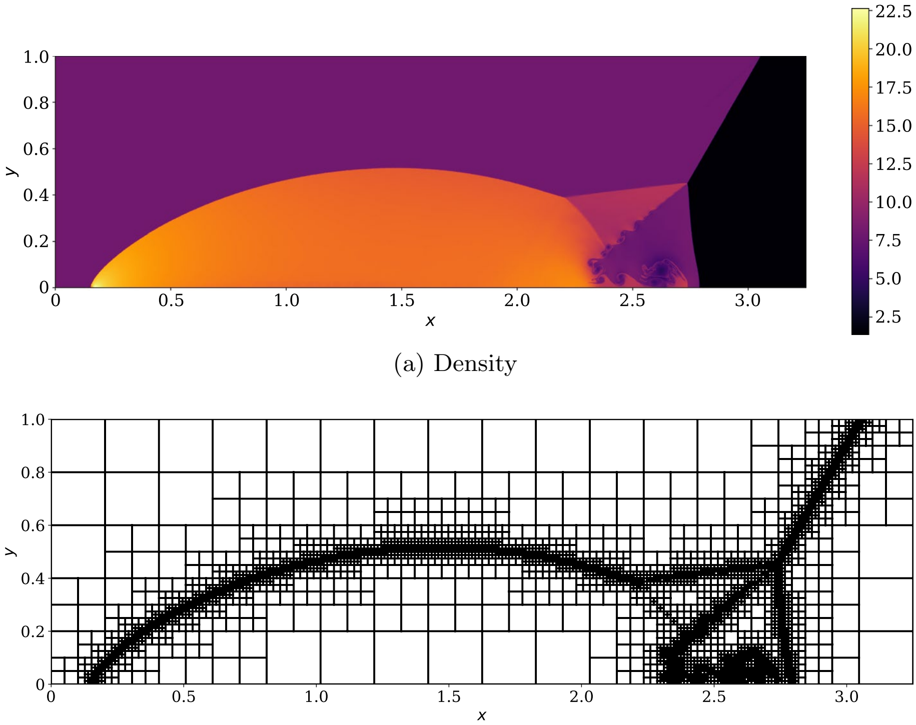
Figure 7 shows the evolution of the time step sizes and the effective CFL number. There is an initial transient period of ca. 100 time steps where both the time step size and the effective CFL number are much smaller than in the remaining simulation. This small initial step size is selected by the automatic detection algorithm and the error-based step size control, and no manual intervention is necessary. The same simulation setup run with the CFL-based step size control crashes in the first few time steps, even with a small CFL number of $\nu = 0.01$. The initial transient period can be captured with even smaller CFL numbers, but running the simulation afterwards in a reasonable amount of time required adapting the CFL factor. In contrast, the error-based step size control can be used with the same parameters throughout the simulation.

5.2 Astrophysical Mach 2 000 Jet

Next, we consider a simulation of an astrophysical Mach 2 000 jet using the ideal compressible Euler equations with ratio of specific heat $\gamma = 5/3$ based on the setup of [60]. Specifically, we use the initial condition

$$\rho^0 = 0.5, \quad v_1^0 = 0, \quad v_2^0 = 0, \quad p^0 = 0.417\,2 \tag{14}$$

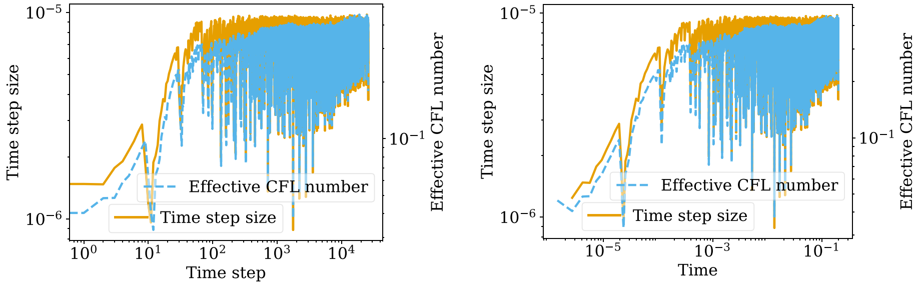
in the domain $[-0.5, 0.5]$ equipped with periodic boundary conditions in the y-direction and Dirichlet boundary conditions using the initial data in the x-direction except for $t > 0$, $x =- 0.5$, and $y \in [- 0.05, 0.05]$, where we use boundary data (denoted by a superscript b)



(a) Density

(b) Mesh

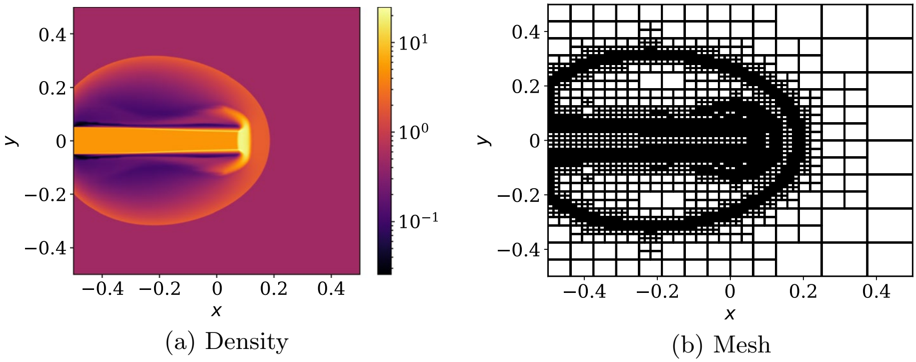
Fig. 6 Density and mesh at the final time of the simulation of the Woodward-Colella double Mach reflection using SSP3(2)4[3S$^*_+$] applied to entropy-dissipative shock capturing semi-discretizations with positivity-preserving limiters and adaptive mesh refinement



(a) Dependence on time step number

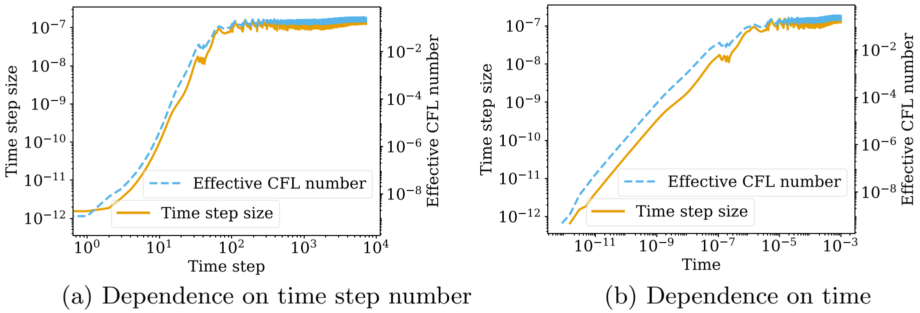
(b) Dependence on time

Fig. 7 Time step sizes and effective CFL numbers of SSP3(2)4[3S$^*_+$] applied to entropy-dissipative shock capturing semi-discretizations with positivity-preserving limiters and adaptive mesh refinement for the double Mach reflection problem



(a) Density (b) Mesh

Fig. 8 Density and mesh at the final time of the simulation of an astrophysical Mach 2 000 jet using SSP3(2)4[3S*$_+$] applied to entropy-dissipative shock capturing semi-discretizations with positivity-preserving limiters and adaptive mesh refinement



(a) Dependence on time step number (b) Dependence on time

Fig. 9 Time step sizes and effective CFL numbers of SSP3(2)4[3S*$_+$] applied to entropy-dissipative shock capturing semi-discretizations with positivity-preserving limiters and adaptive mesh refinement for the astrophysical Mach 2 000 jet

$$\rho^{\mathrm{b}} = 5, \quad v_1^{\mathrm{b}} = 800, \quad v_2^{\mathrm{b}} = 0, \quad p^{\mathrm{b}} = 0.417\,2. \tag{15}$$

We use an initially uniform mesh with 64 elements per coordinate direction and polynomials of degree $p = 3$. We use the shock capturing approach of [42] with the entropy-conservative numerical flux of [75, 80, 81] in the volume terms and a local Lax-Friedrichs flux at interfaces and finite volume subcells. After every time step, we apply adaptive mesh refinement. Moreover, we apply the positivity-preserving limiter of [99] for density and pressure after every RK stage. We integrate the semi-discrete system in time using SSP3(2)4[3S*$_+$] for $t \in [0, 10^{-3}]$. The complete setup can be found in the reproducibility repository [79].

Figure 8 shows a snapshot of the numerical solution and the adapted grid at the final time $t = 10^{-3}$. While the initial condition is given by uniform free stream values, the boundary condition (15) changes the numerical approximation rapidly in time. Consequently, the mesh is refined over time and adapts to the solution structures.

Figure 9 shows the evolution of the time step for this demanding simulation. The initial time step must be very small—of the order $\Delta t \approx 10^{-12}$—to allow the simulation to run. For this setup, we set $\Delta t = 10^{-12}$ initially; other choices are also possible and are adapted to similar values automatically by the controller, but the fully automatic initial guess is not sufficient in this case.

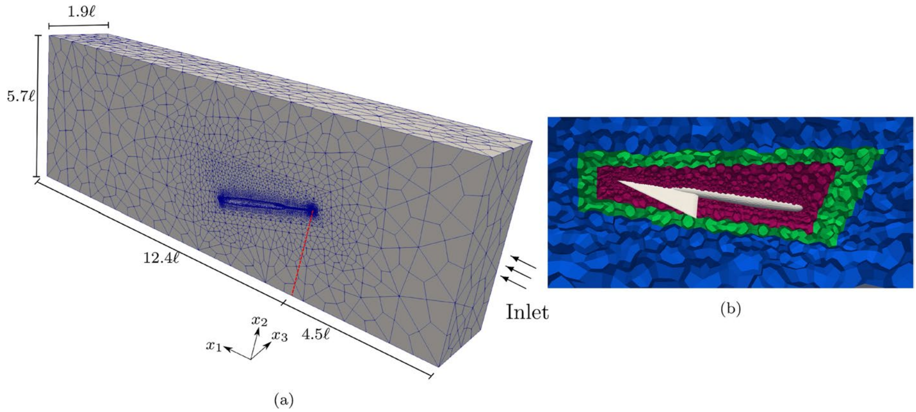


Fig. 10 Solution polynomial degree distribution, p, computational domain and boundary mesh elements for the 65° swept delta wing test case [67]

The initial transient period lasts for ca. 100 time steps where the error-based step size controller increases the time step size gradually. Afterwards, the time step size plateaus. Handling such an initial transient period is difficult for standard CFL-based step size controllers. Initially, a CFL factor of $\nu \approx 10^{-9}$ is required; $\nu = 10^{-8}$ results in a blow-up of the simulation. One could implement a more complicated CFL-based controller adapting the CFL factor in the initial transient period or restart the simulation after ca. 100 time steps with a larger CFL factor. However, no such additional techniques are required for the error-based step size control, making it easier to use in this case.

5.3 Delta Wing

The next test case setup approximates the solution of the compressible Navier-Stokes equations. We consider an industrially relevant simulation of a 65° swept delta wing at a Mach number $Ma = 0.9$, a Reynolds number $Re = 10^6$ (based on the mean aerodynamic chord), and setting the angle of attack to AoA = 13°. The geometry was proposed by Hummel and Redeker [47] for the Second International Vortex Flow Experiment (VFE-2) and manufactured based on an NASA wing geometry that served as reference configuration [20]. We consider the flow past the leading-edge configuration with a medium radius leading edge $r_{\mathrm{LE}}/\bar{c} = 0.001\,5$, where $\bar{c} = 0.653\,\mathrm{m}$. The delta wing has a mean aerodynamic chord of $\ell = 0.667\,\mathrm{m}$, a root chord length of $c_{\mathrm{r}} = 1.47\ell$, and a wing span of $b = 1.37\ell$. The sting present in the wind tunnel testing is kept downstream as part of the geometry up to the position $x_1/c_{\mathrm{r}} = 1.758$, where the x_1 Cartesian coordinate points in the streamwise direction, as shown in the left part of Fig. 10.

For this test, we use the *hp*-adaptive entropy-stable solver SSDC [67], which is able to deliver numerical results in good agreement with experimental data. Entropy-stable adiabatic no-slip wall boundary conditions [23] are applied to the wing and sting surfaces, whereas freestream far-field boundary conditions are applied at the inlet and outlet planes. Due to the symmetry of the problem in the span-wise direction, half span of the flow is modeled through a symmetry boundary condition. On the rest of the boundary planes, entropy-stable inviscid wall boundary conditions [69] are prescribed. The

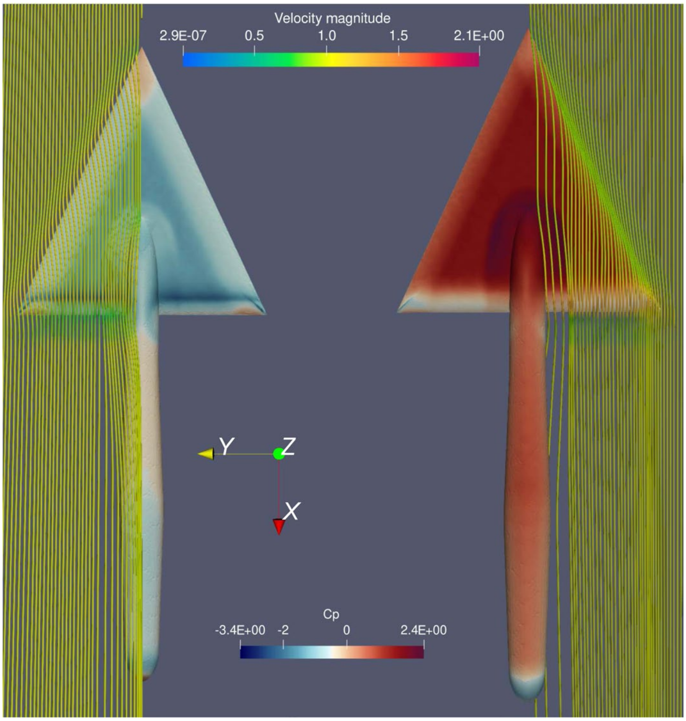


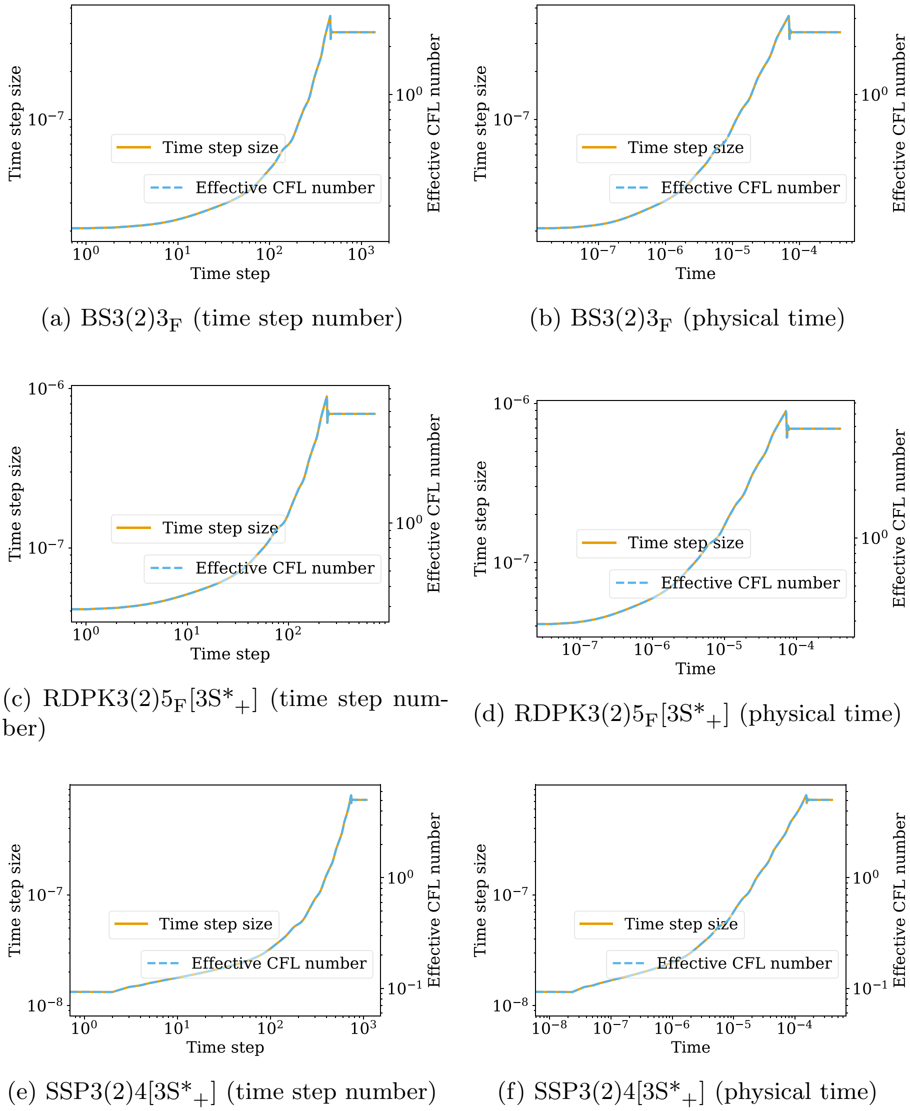
Fig. 11 Instantaneous pressure coefficient distribution and streamlines colored by the velocity magnitude at $t = 0.05$ for the 65° swept delta wing test case [67] (left: top view; right: bottom view)

grid consists of $\approx 9.2 \times 10^4$ hexahedral elements. It is subdivided into three blocks, as shown in the right part of Fig. 10, with each block corresponding to a different solution polynomial degree p. In particular, we use $p = 2$ in the far-field region, $p = 5$ in the region surrounding the delta wing and its support, and $p = 3$ elsewhere. Given the degree of the solution and the number of elements in each block, the total number of degrees of freedom (DOFs) is $\approx 1.435 \times 10^7$.

The flow around a delta wing is peculiar. When the angle of attack exceeds 7°, typically, flow separation occurs at the leading edge. The roll-up of the leading-edge vortices induces low pressure on the upper surface of the wing and enhances the lift. Figure 11 shows the instantaneous pressure coefficient distribution and streamlines colored by the velocity magnitude at $t = 0.05$.

Results of simulations using some representative RK methods are shown in Fig. 12. For this cold start of the simulation with free stream values, all methods begin with a conservative estimate of the time step size. The error-based step size controllers increase the time step size in the first few time steps and reach an asymptotically constant step size after a few hundred time steps.

The effective CFL number shown in Fig. 12 is basically proportional to the time step size itself. Running SSDC with the CFL-based step size control requires a CFL factor that is (at least initially)



(a) BS3(2)3_F (time step number)

(b) BS3(2)3_F (physical time)

(c) RDPK3(2)5_F[3S*$_+$] (time step number)

(d) RDPK3(2)5_F[3S*$_+$] (physical time)

(e) SSP3(2)4[3S*$_+$] (time step number)

(f) SSP3(2)4[3S*$_+$] (physical time)

Fig. 12 Effective CFL numbers and time step sizes depending on the time step index (left column) and physical time (right column) of some representative RK methods applied to entropy-dissipative semi-discretizations of the compressible Navier-Stokes equations for the delta wing setup

- 13% smaller for BS3(2)3_F,
- 21% smaller for RDPK3(2)5_F[3S$^*{}_+$],
- 32% smaller for SSP3(2)4[3S$^*{}_+$],

than the asymptotic CFL factor to avoid a blow-up of the simulation. The required number of RHS evaluations and rejected steps are listed in Table 2.

Table 2 Performance of representative RK methods with error-based step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for the delta wing setup up to time $t = 0.015$

Scheme	β	`Tol`	#FE	#A	#R
BS3(2)3$_F$	(0.60, – 0.20, 0.00)	10^{-8}	17 641	5 875	4
RDPK3(2)5$_F$[3S$^*_+$]	(0.70, – 0.23, 0.00)	10^{-8}	14 963	2 989	3
SSP3(2)4[3S$^*_+$]	(0.55, – 0.27, 0.05)	10^{-8}	13 392	3 344	4

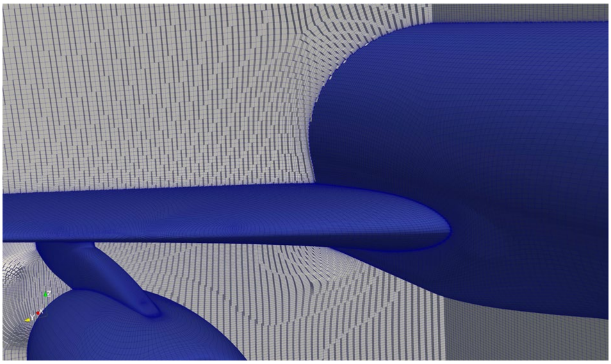


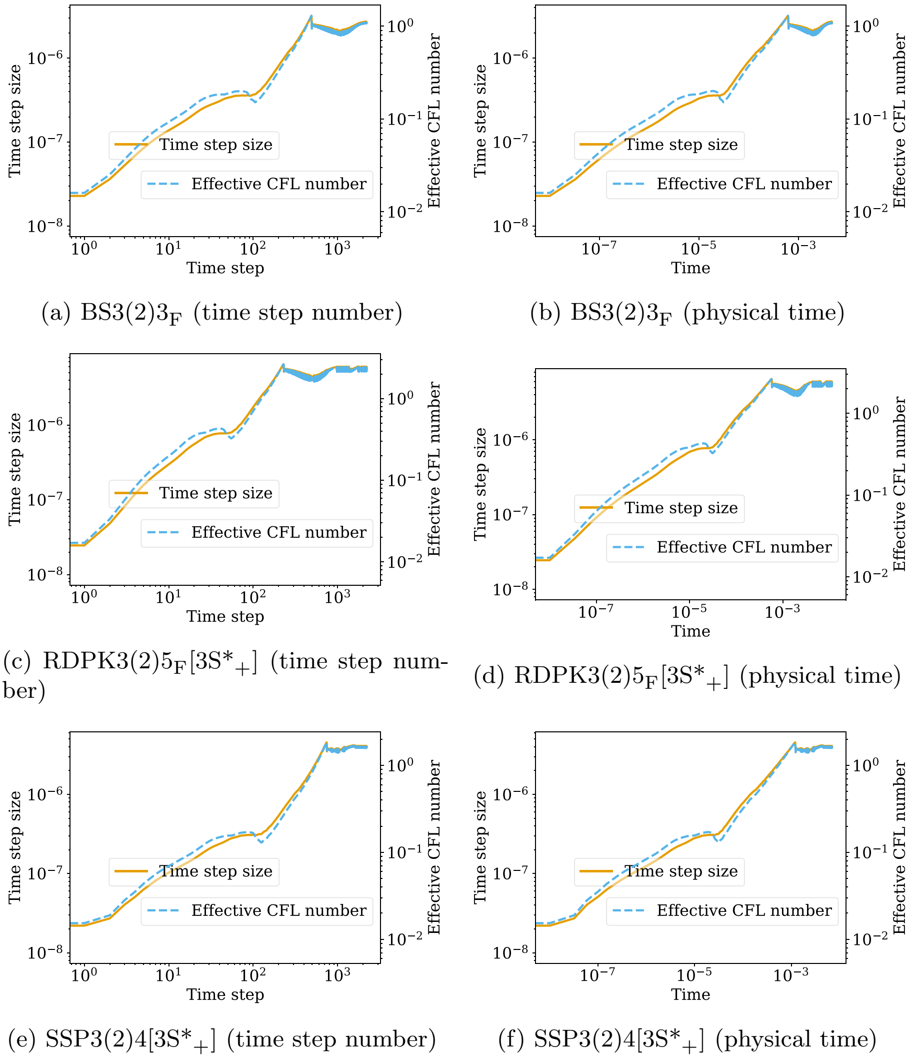
Fig. 13 Zoom of the mesh for the NASA CRM generated at KAUST

5.4 NASA Common Research Model

The NASA common research model (CRM) was conceived in 2007. Its aerodynamic design was completed in 2008, responding to needs broadly expressed within the US and international aeronautics communities for modern/industry-relevant geometries coupled with advanced experimental data for applied computational fluid dynamic validation studies [82]. Here, we consider the compressible Navier-Stokes equations for a flow over the NASA CRM at an angle of attack of 10°, a Mach number $Ma = 0.85$, and a Reynolds number $Re = 5 \times 10^6$.

The computational domain contains 12.8×10^6 hexahedral elements with a maximum aspect ratio of approximately 105. Entropy-stable adiabatic no-slip wall boundary conditions [23] are applied to the aircraft, whereas freestream far-field boundary conditions are applied at the inlet and outlet planes. We set a solution polynomial degree $p = 3$ in the whole domain. Given the degree of the solution and the number of cells, the number of DOFs is $\approx 7.68 \times 10^8$. A zoom of the mesh near the surface of the right wing and the nacelle is shown in Fig. 13.

Results of simulations using some representative RK methods are shown in Fig. 14. Again, all methods use a conservative estimate of the initial time step size Δt, which then increases Δt monotonically in the first few hundred time steps and reaches an asymptotically approximately constant step size afterwards. The effective CFL number shown in Fig. 14 is, again, basically proportional to the time step size itself. Running SSDC with the CFL-based step size control requires a CFL factor that is (at least initially)



(a) BS3(2)3_{F} (time step number)

(b) BS3(2)3_{F} (physical time)

(c) RDPK3(2)5_{F}[3S$^*{}_+$] (time step number)

(d) RDPK3(2)5_{F}[3S$^*{}_+$] (physical time)

(e) SSP3(2)4[3S$^*{}_+$] (time step number)

(f) SSP3(2)4[3S$^*{}_+$] (physical time)

Fig. 14 Effective CFL numbers and time step sizes depending on the time step index (left column) and physical time (right column) of some representative RK methods applied to entropy-dissipative semi-discretizations of the compressible Navier-Stokes equations for the NASA common research model

Table 3 Performance of representative RK methods with error-based step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for the NASA common research model up to time $t = 0.004$

Scheme	β	`Tol`	#FE	#A	#R
BS3(2)3_{F}	(0.60, – 0.20, 0.00)	10^{-8}	5 874	1 958	0
RDPK3(2)5_{F}[3S$^*{}_+$]	(0.70, – 0.23, 0.00)	10^{-8}	4 610	922	0
SSP3(2)4[3S$^*{}_+$]	(0.55, – 0.27, 0.05)	10^{-8}	5 920	1 480	0

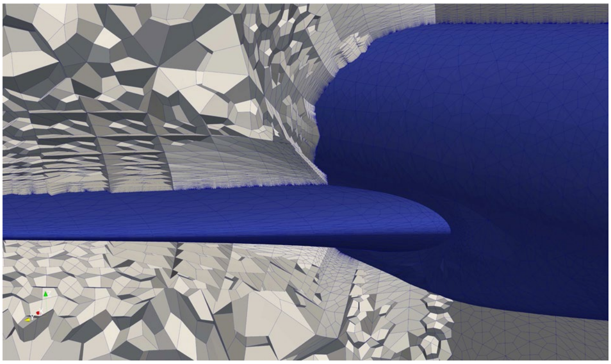


Fig. 15 Zoom of the CRM-WB-a2.75-Coarse-P2A mesh for the NASA CRM provided by the fifth high-order workshop [13]

- 16% smaller for RDPK3(2)5_F[$3S^*_+$],
- 9% smaller for SSP3(2)4[$3S^*_+$],

than the asymptotic CFL factor to avoid a blow-up of the simulation. For BS3(2)3_F, the asymptotic CFL factor from the error-based step size control can be chosen for the complete simulation. The required number of RHS evaluations and rejected steps are listed in Table 3.

To conclude this test case, we highlight that the error-based controller yields a substantially higher level of robustness for this industry-relevant test case under a change of mesh structure. To verify that, we simulate the same flow problem using the mesh illustrated in Fig. 15. This new mesh is one of the grids provided for the fifth high-order workshop held at Ceneaero, Belgium, and it corresponds to the mesh tagged as "CRM-WB-a2.75-Coarse-P2A". When the CFL-based controller is used, the target CFL number is set to the effective asymptotic value obtained from the previous simulations with the mesh shown in Fig. 13, i.e., approximately 3.6 for all three RK methods (see Fig. 14). The simulations run successfully to the final time for all the three time integration methods tested, i.e., BS3(2)3_F, RDPK3(2)5_F[$3S^*_+$], and SSP3(2)4[$3S^*_+$], when the error-based time step controller is used. On the contrary, all simulations fail when we choose the CFL-based time step controller. The reason is simple, the effective asymptotic value of the CFL number for this new mesh is approximately a factor of three smaller than the value set, i.e., ≈ 1.2. This indicates that the CFL-based controller is not sufficiently robust under a change of mesh. In many cases, we could mitigate the instability by choosing a "very small value" for the target CFL number, but this would yield a very inefficient and computationally expensive time advancement of the simulation.

6 Change of Variables: Dissipative Methods Using an Entropy Projection

Whenever a semi-discretization is not strictly a collocation method, there are some transformations between the basic solution variables evolved in time and nodal values required to compute fluxes, e.g., methods using modal coefficients [16, 62], entropy variables as primary unknowns [43, 46], staggered grid methods [56, 68], and DG difference methods [98]. Here, we apply entropy-stable Gauss methods [17] available in Trixi.jl [78, 85] with efficient implementations [77]. While these schemes are more complex than collocation

Table 4 Performance of representative RK methods with default error-based and manually tuned CFL-based step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for the Kelvin-Helmholtz instability with low Atwood number $A = 3/7$ and entropy-dissipative Gauss collocation semi-discretizations using entropy projections

Scheme	β	`Tol`/ν	#FE	#A	#R
BS3(2)3$_{\mathrm{F}}$	(0.60, – 0.20, 0.00)	`Tol` $= 10^{-4}$	7 542	2 511	2
		$\nu = 0.57$	7 801	2 600	
RDPK3(2)5$_{\mathrm{F}}$[3S$^*{}_+$]	(0.70, – 0.23, 0.00)	`Tol` $= 10^{-4}$	6 403	1 278	2
		$\nu = 1.10$	6 856	1 371	
RDPK4(3)9$_{\mathrm{F}}$[3S$^*{}_+$]	(0.38, – 0.18, 0.01)	`Tol` $= 10^{-5}$	6 537	722	4
		$\nu = 1.85$	6 922	769	
SSP3(2)4[3S$^*{}_+$]	(0.55, – 0.27, 0.05)	`Tol` $= 10^{-4}$	5 478	1 367	2
		$\nu = 1.03$	5 672	1 418	

methods due to the entropy projection, they also possess favorable robustness properties for multidimensional simulations of under-resolved compressible flows with variable density and small-scale features [18].

We consider a Kelvin-Helmholtz instability of the 2D compressible Euler equations of an ideal gas with ratio of specific heats $\gamma = 1.4$ as in [18]. The domain $[-1, 1]^2$ is equipped with periodic boundary conditions and the simulation is initialized with

$$\begin{cases} \rho_1 = 1, \quad \rho_2 = \rho_1 \dfrac{1+A}{1-A}, \quad \rho^0 = \rho_1 + B(x,y)(\rho_2 - \rho_1), \\ p^0 = 1, \quad v_1^0 = B(x,y) - \dfrac{1}{2}, \quad v_2^0 = \dfrac{1}{10}\sin(2\pi x), \end{cases} \tag{16}$$

where A is the Atwood number parameterizing the density contrast and

$$B(x,y) = \tanh(15y + 7.5) - \tanh(15y - 7.5) \tag{17}$$

is a smoothed step function. We apply Gauss methods with 32 elements per coordinate direction and polynomials of degree $p = 3$. We choose the entropy-conservative flux of [75, 80, 81] for the volume terms and a local Lax-Friedrichs flux at surfaces. We apply several representative RK methods to evolve the resulting entropy-dissipative semi-discretizations in time $t \in [0, 5]$.

6.1 Mild Initial Conditions with Low Atwood Number

Table 4 shows a summary of the performance statistics of some representative RK methods for a low Atwood number $A = 3/7$. The CFL number ν was maximized manually up to three significant digits (so that the simulations did not crash). We used time integration methods with controllers recommended in [74] and a default tolerance `tol` $= 10^{-4}$; we set `tol` $= 10^{-5}$ for RDPK4(3)9$_{\mathrm{F}}$[3S$^*{}_+$], since the simulation crashed for the looser default tolerance.

The total number of function evaluations and time steps is always smaller with the error-based step size control in this example, usually around 5%. In addition, the

Table 5 Performance of representative RK methods with default error-based and manually tuned CFL-based step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for the Kelvin-Helmholtz instability with high Atwood number $A = 0.7$ and entropy-dissipative Gauss collocation semi-discretizations using entropy projections

Scheme	β	`Tol`/ν	#FE	#A	#R
BS3(2)3$_\mathrm{F}$	(0.60, – 0.20, 0.00)	10^{-4}	9 567	3 185	3
RDPK3(2)5$_\mathrm{F}$[3S$^*_+$]	(0.70, – 0.23, 0.00)	10^{-4}	8 173	1 621	13
RDPK4(3)9$_\mathrm{F}$[3S$^*_+$]	(0.38, – 0.18, 0.01)	10^{-6}	10 587	1 175	1
SSP3(2)4[3S$^*_+$]	(0.55, – 0.27, 0.05)	10^{-4}	6 618	1 640	14

manual optimization of the CFL factor ν required several simulation runs. Note that the optimal CFL factors are different from the ones used for the Orzsag-Tang vortex while the same tolerances were used. For this setup, SSP3(2)4[3S$^*_+$] is the most efficient method, followed by RDPK3(2)5$_\mathrm{F}$[3S$^*_+$].

6.2 Demanding Initial Conditions with High Atwood Number

Increasing the density contrast makes this under-resolved test case more demanding [18]. Violations of positivity are more likely to occur and the sensitivity of the entropy variables for near-vacuum states can cause issues [98, 15]. In particular, the basic variables evolved in time for the Gauss methods violate positivity properties of the density/pressure around $t \approx 3$ even for strict CFL factors such as $\nu = 0.05$ for SSP3(2)4[3S$^*_+$], so that a standard CFL-based step size control is not possible. Similar problems occur for other RK methods.

In contrast, the error-based step size control works directly in most cases, as shown in Table 5. We only needed to use a stricter tolerance `tol` $= 10^{-6}$ for RDPK4(3)9$_\mathrm{F}$[3S$^*_+$]. This RK method has more stages than the other methods, making it more difficult to adapt quickly to changing conditions. Again, SSP3(2)4[3S$^*_+$] is the most efficient method, followed by RDPK3(2)5$_\mathrm{F}$[3S$^*_+$].

7 New Systems: Convenient Exploratory Research

A key feature of error-based time stepping methods is that they require no knowledge of the maximum eigenvalues to select a stable explicit time step. This feature is particularly convenient when one wants to apply the existing numerical methods to new equation systems. To illustrate this, we consider the shallow water equations augmented with an Exner equation to account for interactions between the bottom topography and the fluid flow. The coupled system of equations in two spatial dimensions is

$$\frac{\partial}{\partial t}\begin{pmatrix} h \\ hv_1 \\ hv_2 \\ b \end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix} hv_1 \\ hv_1^2 + \frac{g}{2}h^2 \\ hv_1 v_2 \\ \xi q_1(\boldsymbol{v}) \end{pmatrix} + \frac{\partial}{\partial y}\begin{pmatrix} hv_2 \\ hv_1 v_2 \\ hv_2^2 + \frac{g}{2}h^2 \\ \xi q_2(\boldsymbol{v}) \end{pmatrix} = \begin{pmatrix} 0 \\ -gh\frac{\partial b}{\partial x} \\ -gh\frac{\partial b}{\partial y} \\ 0 \end{pmatrix}, \tag{18}$$

where $h(x, y, t)$ is the water height, $\boldsymbol{v}(x,y,t) = (v_1(x,y,t), v_2(x,y,t))^{\mathrm{T}}$ is the velocity field, $b(x, y, t)$ is the evolving bottom topography, and g is the gravitational constant.

The last line in (18) is the Exner equation for the time evolution of the bottom topography b. It contains the constant $\xi = 1/(1-\sigma)$ where $\sigma \in (0,1)$ is the porosity of the bottom material, and the solid transport discharge $\boldsymbol{q} = (q_1, q_2)^{\mathrm{T}}$ as a function of the flow velocity $\boldsymbol{v}$. A closure model is required to describe how the bottom topography couples to the flow and defines the form of the $\boldsymbol{q}$ terms. This closure depends on the characteristics of the bottom material and flow, e.g., the grain size or Froude number. There exist many closure models for the solid transport discharge proposed and studied (analytically as well as numerically) in the literature; see e.g., [10, 30, 32, 64]. For the present study, we consider the simple and frequently used model due to Grass [32] that models the instantaneous sediment transport as a power law on the velocity field

$$\boldsymbol{q}(\boldsymbol{v}) = A_g \boldsymbol{v}\left(v_1^2 + v_2^2\right)^{\frac{m-1}{2}}. \tag{19}$$

In (19), the constant $A_g \in [0,1]$ accounts for the porosity of the bottom sediment layer as well as the effects of the grain size and is usually determined from experimental data. When A_g is zero, there is no sediment transport and (18) reduces to the standard two-dimensional shallow water equations. The interaction between the bottom and the water flow is weak when A_g is small and strong as A_g approaches one. The factor $m \in [1,4]$ in the Grass model may also be determined from data; however, if one considers odd integer values of m, then (19) can be differentiated and the model remains valid for all values of the velocity field $\boldsymbol{v}$. For the remainder of this discussion, we will take $m = 3$ and the solid transport discharge will take the form

$$\boldsymbol{q}(\boldsymbol{v}) = A_g \boldsymbol{v}\left(v_1^2 + v_2^2\right). \tag{20}$$

For the particular closure model (20), we investigate the flux Jacobian of the governing system of equations (18). This is because the choice of a stable explicit time step under a CFL condition requires knowledge of the fastest wave speed of the underlying system. For this, we compute the flux Jacobian in the x-direction, which incorporates non-conservative terms from the right-hand side of (18), to be

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ gh - v_1^2 & 2v_1 & 0 & gh \\ -v_1 v_2 & v_2 & v_1 & 0 \\ -\frac{3\xi A_g v_1}{h}\left(v_1^2 + v_2^2\right) & \frac{\xi A_g}{h}\left(3v_1^2 + v_2\right) & \frac{2\xi A_g}{h} v_1 v_2 & 0 \end{pmatrix}. \tag{21}$$

Note that the analysis and results in the y-direction are analogous, so we will only consider (21) for this eigenvalue analysis for time step purposes. With the chosen Grass closure model, it has been shown that all the eigenvalues of (21) are real and the governing system (18) is hyperbolic [19, 26]. The eigenvalues of the Jacobian in the x-direction are v_1 and the three roots of the characteristic polynomial

$$\lambda^3 - 2v_1\lambda^2 + (v_1^2 - gh - g\xi A_g(3v_1^2 + v_2^2))\lambda + g\xi A_g v_1 (3v_1^2 + v_2^2) = 0. \tag{22}$$

It is possible, although unwieldy, to apply Cardano's formula to determine the real, distinct roots of (22). However, an accurate estimate to the maximum wave speed from (22) is necessary for the traditional CFL-based time stepping methods.

For many applications of interest, the characteristic flow speed of the water is much faster than the speed of the bottom topography [19, 45, 92]. In particular, for subcritical flow, the value of v_1 is relatively small, so it is possible to use a loose bound of the flow speeds with $v_1 \pm \sqrt{gh}$ where $\sqrt{gh}$ is the surface wave speed. However, this loose bound applied to a CFL time step condition can result in excessive numerical diffusion that acts on the computed (slowly moving) bottom topography.

To approximate the solution of the shallow water equations with an Exner model (18), we use a DG spectral element method in space to create the semi-discretization and apply explicit time integration; see [95] for details. To couple the moving bottom topography $b(x, y, t)$, we use an unsteady approach where the water flow and riverbed are calculated simultaneously. That is, the water flow can be either steady or unsteady and the changes in the bed update are considered to be significant, i.e., the wave speed of the bed-updating equation is of a similar magnitude to the wave speeds of the water flow. For this approach, the system of flow and bottom evolution equations are discretized simultaneously. This high-order DG method applied to the shallow water equations is well balanced for static and subcritical moving water flow [95, 96].

As an exemplary test case, we consider the evolution of a smooth sand hill bottom topography in a uniform flow governed by the shallow water equations. This test case is a two-dimensional numerical simulation that was first proposed and analyzed by de Vriend [24]. It has become a common test case for benchmarking purposes in the moving bottom topography literature, e.g., [5, 19, 26, 45, 92], as it is easy to set up and demonstrates the ability of a numerical scheme to capture the morphology of a bottom topography governed with (18).

It involves the evolution of a conical sand dune in a channel with a non-erodible bottom on a square domain with dimensions 1 000 m × 1 000 m. The initial conditions of the flow variables for this test case given in conservative variables are

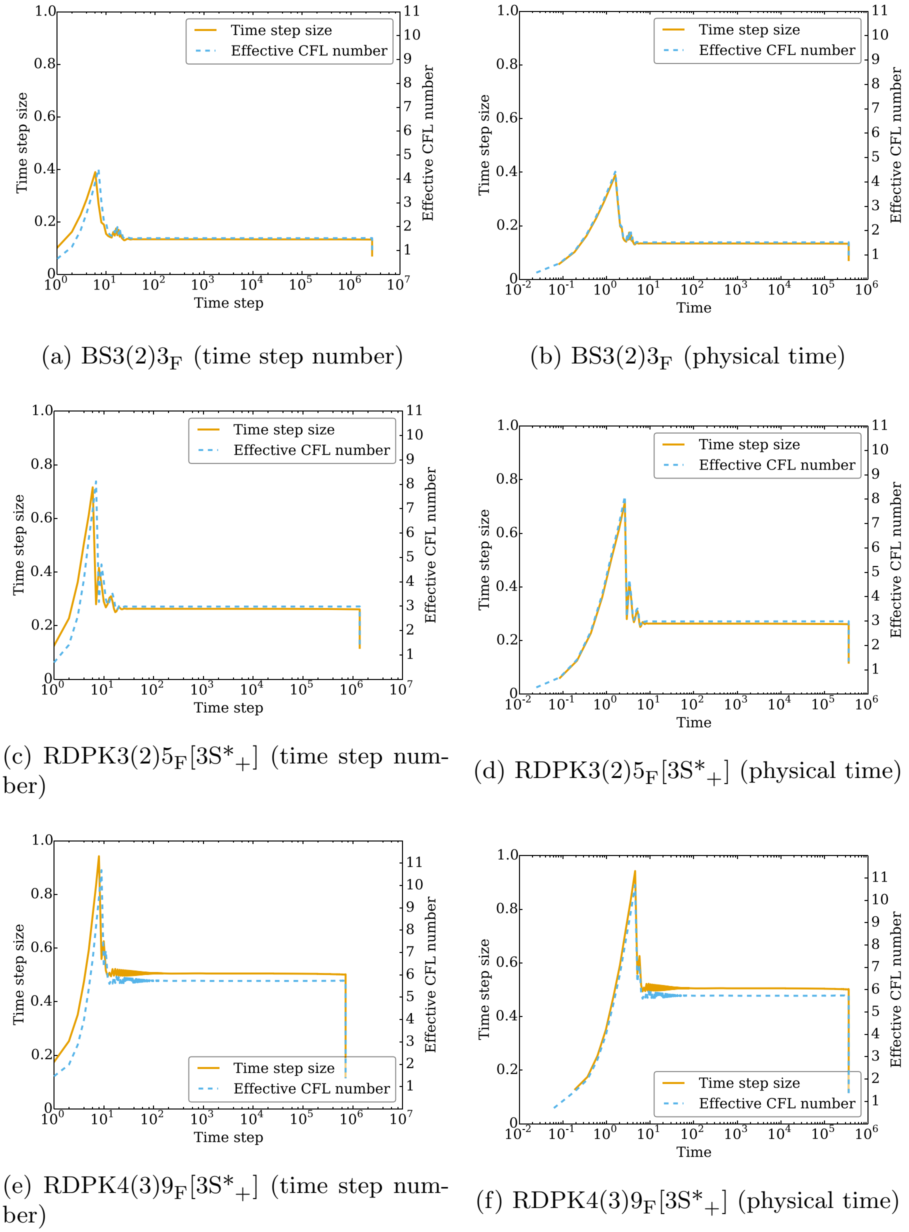
$$\begin{pmatrix} h \\ hv_1 \\ hv_2 \end{pmatrix} = \begin{pmatrix} 10 - b(x, y, 0) \\ 10 \\ 0 \end{pmatrix}, \tag{23}$$

and the initial form of the sediment layer is given by

$$b(x, y, 0) = \begin{cases} \sin^2\left(\frac{(x-300)\pi}{200}\right)\sin^2\left(\frac{(y-400)\pi}{200}\right) & \text{if } 300 \leqslant x \leqslant 500,\ 400 \leqslant y \leqslant 600, \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

We set the gravitational constant to be $g = 9.8\,\mathrm{m/s^2}$. The porosity of the sediment material is $\sigma = 0.4$ and the free parameter from the Grass model (20) is $A_g = 0.001$. The parameter A_g corresponds to the coupling strength and speed of interaction between the sediment and the water flow [5, 45]. Taking a small value for A_g models, weak coupling and a simulation must be run for a longer time period to observe significant variations in the sediment layer. Therefore, we take the final time of this simulation to be $t = 100$ h (360 000 s). The discharge value $hv_1 = 10\ \mathrm{m^2/s}$ from (23) gives the Froude number

$$Fr = \frac{\sqrt{v_1^2 + v_2^2}}{\sqrt{gh}} \approx 0.1. \tag{25}$$



(a) BS3(2)3_F (time step number)

(b) BS3(2)3_F (physical time)

(c) RDPK3(2)5_F[3S$^*_+$] (time step number)

(d) RDPK3(2)5_F[3S$^*_+$] (physical time)

(e) RDPK4(3)9_F[3S$^*_+$] (time step number)

(f) RDPK4(3)9_F[3S$^*_+$] (physical time)

Fig. 16 Time step sizes and effective CFL numbers for three error-based RK methods. Left column: # time steps is the x-axis. Right column: time is the x-axis

Thus, the flow is in the subcritical regime and we must set boundary conditions accordingly; see, e.g., [65]. For this test case, the boundary conditions are taken to be walls on the top and bottom of the domain, the left is a subcritical inflow, and the right is a subcritical outflow. The complete setup can be found in the reproducibility repository [79].

Table 6 Performance of representative RK methods with default error-based and step size controllers: number of function evaluations (#FE), accepted steps (#A), and rejected steps (#R) for a shallow water flow over an evolving sand dune

Scheme	β	$\texttt{tol}/\nu$	#FE	#A	#R
$\mathrm{BS3(2)3_F}$	(0.60, – 0.20, 0.00)	10^{-7}	8 088 687	2 696 226	2
$\mathrm{RDPK3(2)5_F[3S^*{}_+]}$	(0.70, – 0.23, 0.00)	10^{-7}	6 873 513	1 374 701	1
$\mathrm{RDPK4(3)9_F[3S^*{}_+]}$	(0.38, – 0.18, 0.01)	10^{-7}	6 442 851	715 871	1

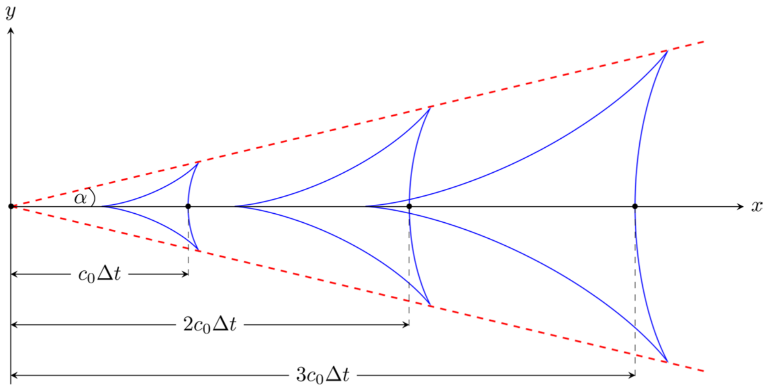
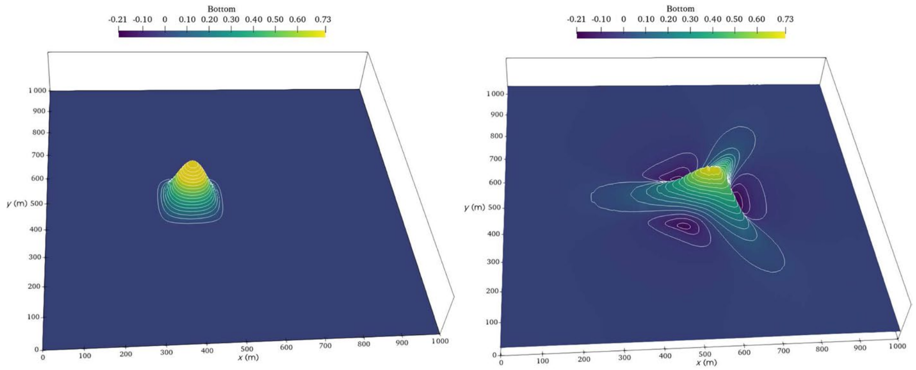


Fig. 17 Characteristic evolution of a dune-shaped sediment distribution

For the spatial discretization, we divide the domain $\varOmega = [0, 1\,000]^2$ into 900 Cartesian elements. In each element, we use polynomials of degree $p = 5$ in each spatial direction for the DG approximation. This results in 32 400 DOFs for each equation in (18). The classic shallow water equations for water height, momentum, and bottom topography terms are discretized with the well-balanced, high-order DG method described by Wintermeyer et al. [95]. The additional Exner equation for sediment transport is discretized with a standard DG spectral element approximation.

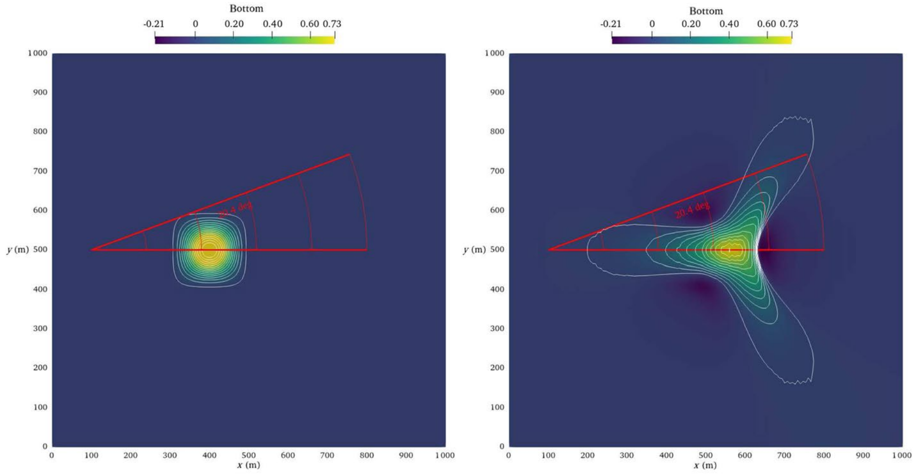
We first investigate the use of the error-based time step control for this shallow water with sediment transport problem setup. We recorded the time step sizes and effective CFL numbers after every accepted time step for three representation RK methods. The tolerance is set to $\texttt{tol} = 10^{-7}$ for all error-based time stepping methods considered. All three time integration techniques work directly for this test case. The behavior of the time step size and the effective CFL number are presented in Fig. 16. Initially, the time step size of the three methods increases before the time step settles to a stable value as the physical problem becomes a quasi-steady flow where the sediment slowly evolves in the presence of the background flow.

We further investigate the effectiveness of the three time integrators with the error-based step size control for this test case, as shown in Table 6. For this quasi-steady-state configuration, we found that a stricter tolerance $\texttt{tol} = 10^{-7}$ for all the integration techniques tested performed the best in the sense of requiring a lower number of time steps. Here, we find that $\mathrm{RDPK4(3)9_F[3S^*{}_+]}$ is the most efficient method, followed by $\mathrm{RDPK3(2)5_F[3S^*{}_+]}$.



(a) $t = 0\,\mathrm{s}$ (b) $t = 360\,000\,\mathrm{s}$

Fig. 18 Projected lateral view of the approximate solution of the bottom topography $b(x, y, t)$ that demonstrates expected star-shaped structure of the conical sand dune at the final time



(a) $t = 0\,\mathrm{s}$ (b) $t = 360\,000\,\mathrm{s}$

Fig. 19 Estimation of the spreading angle $\alpha_{\text{num}} = 20.4°$ for the numerical approximation of a conical sand dune found by examining the iso-level 0.012 5 at the initial and final times. The theoretical prediction of the spreading angle for this problem setup is 21.787°

We examine the expected solution behavior as well as simulation results for the sediment transport problem under consideration. As time evolves, the conical dune (24) gradually spreads into a star-shaped pattern. That is, the sediment will redistribute itself along a triangular shape with a particular spreading angle α and the bulk shape of the dune will move with a speed c_0 as it spreads [24, 45]. This process is illustrated in Fig. 17.

When the Grass model is used and the interaction between the flow and sediment is weak, that is $A_g < 0.01$, de Vriend [24] derived an analytical approximation for this spreading angle. In particular, when $m = 3$ in the Grass model (20), this spreading angle of a conical dune is

$$\alpha = \tan^{-1}\left(\frac{3\sqrt{3}}{13}\right) \approx 21.787^\circ. \tag{26}$$

Next, we present simulation results obtained using RDPK4(3)9_F[$3\mathrm{S}^*{}_+$], which was found to be the most efficient. The numerical results from the other two time integration techniques were similar and will not be presented. Figure 18 shows the bottom topography at the initial and final times. From this figure, we see that the discretization captures the expected star-shaped pattern of the sediment evolution.

We also estimate the spreading angle of a conical sand dune obtained for the numerical simulation [92]. This is done by examining the angle up to the iso-level 0.012 5 over time, as shown in Fig. 19. We are able to numerically estimate the spreading angle to be $\alpha_\mathrm{num} = 20.4^\circ$. The numerical spreading angle of the current method compares well to the theoretical angle (26) as well as to similar angle studies done in the literature [5, 26, 45, 92].

We applied error-based time stepping methods to the shallow water equations augmented with an Exner equation to model sediment transport. This served to demonstrate that it is convenient and straightforward to extend the existing code for a new system of equations and obtain a first set of numerical results. This was particularly useful for the considered shallow water Exner equations, because eigenvalue estimates for a general problem setup are unwieldy to obtain [19, 26]. These preliminary results for the evolution and spreading of a conical sand dune compared well to the theoretical solution behavior as well as results from the literature. The development of high-order DG methods for sediment transport problems is the subject of ongoing research. Details about the physical problem setup and numerical parameters as well as all code necessary to reproduce the figures in this section are available in the reproducibility repository for this article [79].

8 Summary and Conclusions

We have compared CFL- and error-based step size controls of explicit RK methods applied to DG semi-discretizations of compressible CFD problems. Our numerical experiments demonstrate that the error-based step size control is convenient and robust in a wide range of applications, including industrially relevant complex geometries with curved meshes in multiple dimensions, shock capturing schemes, initial transient periods (cold start of a simulation), semi-discretizations involving a change of variables, and for exploratory research involving new systems of equations.

There are cures for some problems of the CFL-based step size control discussed in this article. For example, a cold start requiring an initially smaller CFL factor could run a few hundred time steps with a small CFL factor and use a larger CFL factor after a restart—or increase the CFL factor based on a user-supplied function in the first time steps. A change of linear stability behavior of shock capturing schemes could be handled by including the shock capturing indicator in the CFL-based step size control. However, this requires sophisticated additional mechanisms as implemented in some in-house codes. On the other hand, the error-based step size control worked right out of the box without further modifications for the problems considered here. It does not require special handling of many edge cases and comes with a reduced sensitivity with respect to user-supplied parameters.

While we have shown that the error-based step size control is convenient and robust in many applications, including industrially relevant large-scale computations of compressible CFD problems, it does not come with hard mathematical guarantees. For example, some invariant domain preserving methods come with some robustness properties guaranteed under certain CFL constraints; the error-based step size control alone can usually not be proven to satisfy these constraints in all cases. Nevertheless, based on our experience, it is usually robust for suitably chosen shock capturing mechanisms.

As a byproduct of this research, we also compared several time integration methods for hyperbolic conservation laws and compressible flow problems. For lower Mach number flows or problems with positivity issues and/or strong shock capturing requirements, SSP3(2)4[$3S^*{}_+$] is usually the most efficient scheme. Otherwise, RDPK3(2)5_{F}[$3S^*_+$] and RDPK4(3)9_{F}[$3S^*{}_+$] are usually the most efficient methods, where RDPK3(2)5_{F}[$3S^*{}_+$] tends to be more robust and RDPK4(3)9_{F}[$3S^*{}_+$] tends to be more efficient for stricter tolerances. Nevertheless, BS3(2)3_{F} is still one of the best general-purpose ODE solvers for problems like the ones considered in this article.

Future research includes the investigation of the error-based step size control with adaptive mesh refinement and stabilization techniques such as positivity-preserving limiters [99]. These are usually implemented as callbacks outside of the inner time integration loop and are thus “not seen” by error estimators. Moreover, we would like to optimize new RK methods including error estimators and step size controllers for compressible CFD problems in other ranges than [2], e.g., focusing on lower Mach flows and dissipative effects.

Funding Open Access funding enabled and organized by Projekt DEAL. Andrew Winters was funded through Vetenskapsrådet, Sweden Grant Agreement 2020-03642 VR. Some computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC) at Tetralith, partially funded by the Swedish Research Council under Grant Agreement No. 2018-05973. Hugo Guillermo Castro was funded through the award P2021-0004 of King Abdullah University of Science and Technology. Some of the simulations were enabled by the Supercomputing Laboratory and the Extreme Computing Research Center at King Abdullah University of Science and Technology. Gregor Gassner acknowledges funding through the Klaus-Tschira Stiftung via the project “HiFiLab”. Gregor Gassner and Michael Schlottke-Lakemper acknowledge funding from the Deutsche Forschungsgemeinschaft through the research unit “SNuBIC” (DFG-FOR5409).

Data Availability The source code and datasets generated and analyzed for this study using open source codes are available in [79].

Compliance with Ethical Standards

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Abhyankar, S., Brown, J., Constantinescu, E.M., Ghosh, D., Smith, B.F., Zhang, H.: PETSc/TS: A Modern Scalable ODE/DAE Solver Library. 1806.01437 (2018)
2. Al Jahdali, R., Boukharfane, R., Dalcin, L., Parsani, M.: Optimized explicit Runge-Kutta schemes for entropy stable discontinuous collocated methods applied to the Euler and Navier-Stokes equations. In: AIAA Scitech 2021 Forum, AIAA 2021-0633 (2021). https://doi.org/10.2514/6.2021-0633
3. Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W.D., Kaushik, D., Knepley, M.G., May, D.A., McInnes, L.C., Mills, R.T., Munson, T., Rupp, K., Sanan, P., Smith, B.F., Zampini, S., Zhang, H., Zhang, H.: PETSc users manual. Technical Report ANL-95/11 - Revision 3.13, Argonne National Laboratory (2020)
4. Becker, T., Burovskiy, P., Nestorov, A.M., Palikareva, H., Reggiani, E., Gaydadjiev, G.: From exaflop to exaflow. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017, pp. 404–409. European Design and Automation Association, Leuven, IEEE (2017)
5. Benkhaldoun, F., Sahmim, S., Seaid, M.: A two-dimensional finite volume morphodynamic model on unstructured triangular grids. Int. J. Numer. Methods Fluids **63**(11), 1296–1327 (2010). https://doi.org/10.1002/fld.2129
6. Berzins, M.: Temporal error control for convection-dominated equations in two space dimensions. SIAM J. Sci. Comput. **16**(3), 558–580 (1995). https://doi.org/10.1137/0916036
7. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: a fresh approach to numerical computing. SIAM Rev. **59**(1), 65–98 (2017). https://doi.org/10.1137/141000671
8. Bogacki, P., Shampine, L.F.: A 3(2) pair of Runge-Kutta formulas. Appl. Math. Lett. **2**(4), 321–325 (1989). https://doi.org/10.1016/0893-9659(89)90079-7
9. Bohm, M., Winters, A.R., Gassner, G.J., Derigs, D., Hindenlang, F., Saur, J.: An entropy stable nodal discontinuous Galerkin method for the resistive MHD equations. Part I: theory and numerical verification. J. Comput. Phys. **422**, 108076 (2020). https://doi.org/10.1016/j.jcp.2018.06.027
10. Briganti, R., Dodd, N., Kelly, D., Pokrajac, D.: An efficient and flexible solver for the simulation of the morphodynamics of fast evolving flows on coarse sediment beaches. Int. J. Numer. Methods Fluids **69**(4), 859–877 (2012). https://doi.org/10.1002/fld.2618
11. Burstedde, C., Wilcox, L.C., Ghattas, O.: p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. SIAM J. Sci. Comput. **33**(3), 1103–1133 (2011). https://doi.org/10.1137/100791634
12. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations. John Wiley & Sons Ltd, Chichester (2016)
13. Cenaero: HiOCFD5, 5th International Workshop on High-Order CFD Methods (2018). https://how5.cenaero.be
14. Chalmers, N., Krivodonova, L.: A robust cfl condition for the discontinuous Galerkin method on triangular meshes. J. Comput. Phys. **403**, 109095 (2020). https://doi.org/10.1016/j.jcp.2019.109095
15. Chan, J.: On discretely entropy conservative and entropy stable discontinuous Galerkin methods. J. Comput. Phys. **362**, 346–374 (2018). https://doi.org/10.1016/j.jcp.2018.02.033
16. Chan, J.: Skew-symmetric entropy stable modal discontinuous Galerkin formulations. J. Sci. Comput. **81**(1), 459–485 (2019). https://doi.org/10.1007/s10915-019-01026-w
17. Chan, J., Fernández, D.C.D.R., Carpenter, M.H.: Efficient entropy stable Gauss collocation methods. SIAM J. Sci. Comput. **41**(5), 2938–2966 (2019). https://doi.org/10.1137/18M1209234
18. Chan, J., Ranocha, H., Rueda-Ramirez, A.M., Gassner, G.J., Warburton, T.: On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows. Front. Phys. (2022). https://doi.org/10.3389/fphy.2022.898028
19. Chertock, A., Kurganov, A., Wu, T.: Operator splitting based central-upwind schemes for shallow water equations with moving bottom topography. Commun. Math. Sci. **18**(8), 2149–2168 (2020). https://doi.org/10.4310/CMS.2020.v18.n8.a3
20. Chu, J., Luckring, J.M.: Experimental surface pressure data obtained on 65° delta wing across Reynolds number and Mach number ranges. NASA-TM-4645 (1996)
21. Conde, S., Fekete, I., Shadid, J.N.: Embedded pairs for optimal explicit strong stability preserving Runge-Kutta methods. J. Comput. Appl. Math. **412**, 114325 (2022). https://doi.org/10.1016/j.cam.2022.114325
22. Courant, R., Friedrichs, K.O., Lewy, H.: Über die partiellen Differenzengleichungen der mathematischen Physik. Mathematische Annalen **100**(1), 32–74 (1928). https://doi.org/10.1007/BF01448839
23. Dalcin, L., Rojas, D., Zampini, S., Fernández, D.C.D.R., Carpenter, M.H., Parsani, M.: Conservative and entropy stable solid wall boundary conditions for the compressible Navier-Stokes equations:

Adiabatic wall and heat entropy transfer. J. Comput. Phys. **397**, 108775 (2019). https://doi.org/10.1016/j.jcp.2019.06.051

24. De Vriend, H.: 2DH mathematical modelling of morphological evolutions in shallow water. Coastal Eng. **11**(1), 1–27 (1987). https://doi.org/10.1016/0378-3839(87)90037-8
25. Derigs, D., Winters, A.R., Gassner, G.J., Walch, S., Bohm, M.: Ideal GLM-MHD: about the entropy consistent nine-wave magnetic field divergence diminishing ideal magnetohydrodynamics equations. J. Comput. Phys. **364**, 420–467 (2018). https://doi.org/10.1016/j.jcp.2018.03.002
26. Diaz, M.C., Fernández-Nieto, E.D., Ferreiro, A., Parés, C.: Two-dimensional sediment transport models in shallow water equations. A second order finite volume approach on unstructured meshes. Comput. Methods Appl. Mech. Eng. **198**(33/34/35/36), 2520–2538 (2009). https://doi.org/10.1016/j.cma.2009.03.001
27. Dormand, J.R., Prince, P.J.: A family of embedded Runge-Kutta formulae. J. Comput. Appl. Math. **6**(1), 19–26 (1980). https://doi.org/10.1016/0771-050X(80)90013-3
28. Dumbser, M., Zanotti, O., Loubère, R., Diot, S.: A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. J. Comput. Phys. **278**, 47–75 (2014). https://doi.org/10.1016/j.jcp.2014.08.009
29. Fisher, T.C., Carpenter, M.H.: High-order entropy stable finite difference schemes for nonlinear conservation laws: finite domains. J. Comput. Phys. **252**, 518–557 (2013). https://doi.org/10.1016/j.jcp.2013.06.014
30. Garres-Diaz, J., Diaz, M.J.C., Koellermeier, J., de Luna, T.M.: Shallow water moment models for bedload transport problems. arXiv: 2008.08449 (2020). https://doi.org/10.48550/arXiv.2008.08449
31. Gassner, G.J., Winters, A.R., Kopriva, D.A.: Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. J. Comput. Phys. **327**, 39–66 (2016). https://doi.org/10.1016/j.jcp.2016.09.013
32. Grass, A.: Sediment transport by waves and currents. Technical Report FL29, SERC London Centre for Marine Technology (1981)
33. Guermond, J.-L., Nazarov, M., Popov, B., Tomas, I.: Second-order invariant domain preserving approximation of the Euler equations using convex limiting. SIAM J. Sci. Comput. **40**(5), 3211–3239 (2018). https://doi.org/10.1137/17M1149961
34. Guermond, J.-L., Pasquetti, R., Popov, B.: Entropy viscosity method for nonlinear conservation laws. J. Comput. Phys. **230**(11), 4248–4267 (2011). https://doi.org/10.1016/j.jcp.2010.11.043
35. Guermond, J.-L., Kronbichler, M., Maier, M., Popov, B., Tomas, I.: On the implementation of a robust and efficient finite element-based parallel solver for the compressible Navier-Stokes equations. Comput. Methods Appl. Mech. Eng. **389**, 114250 (2022)
36. Gustafsson, K.: Control theoretic techniques for stepsize selection in explicit Runge-Kutta methods. ACM Trans. Math. Softw. (TOMS) **17**(4), 533–554 (1991). https://doi.org/10.1145/210232.210242
37. Gustafsson, K., Lundh, M., Söderlind, G.: A PI stepsize control for the numerical solution of ordinary differential equations. BIT Num. Math. **28**(2), 270–287 (1988). https://doi.org/10.1007/BF01934091
38. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I: Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78862-1
39. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-05221-7
40. Hall, G., Higham, D.J.: Analysis of stepsize selection schemes for Runge-Kutta codes. IMA J. Numer. Analysis **8**(3), 305–310 (1988). https://doi.org/10.1093/imanum/8.3.305
41. Heinisch, P., Ostaszewski, K., Ranocha, H.: Towards green computing: a survey of performance and energy efficiency of different platforms using OpenCL. In: Proceedings of the International Workshop on OpenCL. IWOCL'20, April 2020, Munich (Germany). ACM, New York, NY, USA (2020). https://doi.org/10.1145/3388333.3403035
42. Hennemann, S., Rueda-Ramirez, A.M., Hindenlang, F.J., Gassner, G.J.: A provably entropy stable subcell shock capturing approach for high order split form DG for the compressible Euler equations. J. Comput. Phys. **426**, 109935 (2021). https://doi.org/10.1016/j.jcp.2020.109935
43. Hiltebrand, A., Mishra, S.: Entropy stable shock capturing space-time discontinuous Galerkin schemes for systems of conservation laws. Numerische Mathematik **126**(1), 103–151 (2014). https://doi.org/10.1007/s00211-013-0558-0
44. Hindenlang, F., Gassner, G.J.: A new entropy conservative two-point flux for ideal MHD equations derived from first principles. In: HONOM 2019: European Workshop on High Order Numerical Methods for Evolutionary PDEs, Theory and Applications (2019)

45. Hudson, J.: Numerical techniques for morphodynamic modelling. PhD thesis, University of Reading, Berkshire, United Kingdom (2001)
46. Hughes, T.J.R., Franca, L.P., Mallet, M.: A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. Comput. Methods Appl. Mech. Eng. **54**(2), 223–234 (1986). https://doi.org/10.1016/0045-7825(86)90127-1
47. Hummel, D., Redeker, G.: A new vortex flow experiment for computer code validation. RTO/AVT Symposium on Vortex Flow and High Angle of Attack Aerodynamics, Meeting Proc. RTO-MP-069, 8–31 (2003)
48. Karniadakis, G.E., Sherwin, S.: Spectral/*hp* Element Methods for Computational Fluid Dynamics. Oxford University Press, Oxford (2013). https://doi.org/10.1093/acprof:oso/9780198528692.001.0001
49. Kennedy, C.A., Carpenter, M.H., Lewis, R.M.: Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations. Appl. Numer. Math. **35**(3), 177–219 (2000). https://doi.org/10.1016/S0168-9274(99)00141-5
50. Ketcheson, D.I.: Runge-Kutta methods with minimum storage implementations. J. Comput. Phys. **229**(5), 1763–1773 (2010). https://doi.org/10.1016/j.jcp.2009.11.006
51. Ketcheson, D.I., Mortensen, M., Parsani, M., Schilling, N.: More efficient time integration for Fourier pseudo-spectral DNS of incompressible turbulence. Int. J. Num. Methods Fluids **92**(2), 79–93 (2020). https://doi.org/10.1002/fld.4773
52. Knepley, M.G., Karpeev, D.A.: Mesh algorithms for PDE with Sieve I: mesh distribution. Sci. Program. **17**(3), 215–230 (2009). https://doi.org/10.3233/SPR-2009-0249
53. Kolev, T., Fischer, P., Min, M., Dongarra, J., Brown, J., Dobrev, V., Warburton, T., Tomov, S., Shephard, M.S., Abdelfattah, A.: Efficient exascale discretizations: high-order finite element methods. Int. J. High Perform. Comput. Appl. **35**(6), 527–552 (2021). https://doi.org/10.1177/10943420211020803
54. Kopriva, D.A.: Implementing Spectral Methods for Partial Differential Equations: Algorithms for Scientists and Engineers. Springer, New York (2009)
55. Kopriva, D.A., Jimenez, E.: An assessment of the efficiency of nodal discontinuous Galerkin spectral element methods. In: Ansorge, R., Bijl, H., Meister, A., Sonar, T. (eds.) Recent Developments in the Numerics of Nonlinear Hyperbolic Conservation Laws, pp. 223–235. Springer, Berlin (2013)
56. Kopriva, D.A., Kolias, J.H.: A conservative staggered-grid Chebyshev multidomain method for compressible flows. J. Comput. Phys. **125**(1), 244–261 (1996). https://doi.org/10.1006/jcph.1996.0091
57. Kraaijevanger, J.F.B.M.: Contractivity of Runge-Kutta methods. BIT Numer. Math. **31**(3), 482–528 (1991). https://doi.org/10.1007/BF01933264
58. Kubatko, E.J., Dawson, C., Westerink, J.J.: Time step restrictions for Runge-Kutta discontinuous Galerkin methods on triangular grids. J. Comput. Phys. **227**(23), 9697–9710 (2008). https://doi.org/10.1016/j.jcp.2008.07.026
59. Kuzmin, D.: Entropy stabilization and property-preserving limiters for $\mathbb{P}1$ discontinuous Galerkin discretizations of scalar hyperbolic problems. J. Numer. Math. **29**(4), 307–322 (2021). https://doi.org/10.1515/jnma-2020-0056
60. Liu, Y., Lu, J., Shu, C.-W.: An essentially oscillation-free discontinuous Galerkin method for hyperbolic systems. SIAM J. Sci. Comput. **44**(1), 230–259 (2022). https://doi.org/10.1137/21M140835X
61. Maier, M., Kronbichler, M.: Efficient parallel 3D computation of the compressible Euler equations with an invariant-domain preserving second-order finite-element scheme. ACM Trans. Parallel Comput. **8**(3), 1–30 (2021). https://doi.org/10.1145/3470637
62. Meister, A., Ortleb, S., Sonar, T., Wirz, M.: A comparison of the discontinuous-Galerkin- and spectral-difference-method on triangulations using PKD polynomials. J. Comput. Phys. **231**(23), 7722–7729 (2012). https://doi.org/10.1016/j.jcp.2012.07.025
63. Meister, A., Ortleb, S., Sonar, T., Wirz, M.: An extended discontinuous Galerkin and spectral difference method with modal filtering. Zeitschrift für Angewandte Mathematik und Mechanik **93**(6/7), 459–464 (2013). https://doi.org/10.1002/zamm.201200051
64. Meyer-Peter, E., Müller, R.: Formulas for bed-load transport. In: IAHSR 2nd Meeting, Stockholm, Appendix 2, IAH (1948)
65. Nordström, J., Winters, A.R.: A linear and nonlinear analysis of the shallow water equations and its impact on boundary conditions. J. Comput. Phys. **463**, 111254 (2022). https://doi.org/10.1016/j.jcp.2022.111254
66. Orszag, S.A., Tang, C.-M.: Small-scale structure of two-dimensional magnetohydrodynamic turbulence. J. Fluid Mech. **90**(1), 129–143 (1979). https://doi.org/10.1017/S002211207900210X

67. Parsani, M., Boukharfane, R., Nolasco, I.R., Fernández, D.C.D.R., Zampini, S., Hadri, B., Dalcin, L.: High-order accurate entropy-stable discontinuous collocated Galerkin methods with the summation-by-parts property for compressible CFD frameworks: scalable SSDC algorithms and flow solver. J. Comput. Phys. **424**, 109844 (2021). https://doi.org/10.1016/j.jcp.2020.109844
68. Parsani, M., Carpenter, M.H., Fisher, T.C., Nielsen, E.J.: Entropy stable staggered grid discontinuous spectral collocation methods of any order for the compressible Navier-Stokes equations. SIAM J. Sci. Comput. **38**(5), 3129–3162 (2016). https://doi.org/10.1137/15M1043510
69. Parsani, M., Carpenter, M.H., Nielsen, E.J.: Entropy stable wall boundary conditions for the three-dimensional compressible Navier-Stokes equations. J. Comput. Phys. **292**, 88–113 (2015). https://doi.org/10.1016/j.jcp.2015.03.026
70. Parsani, M., Ketcheson, D.I., Deconinck, W.: Optimized explicit Runge-Kutta schemes for the spectral difference method applied to wave propagation problems. SIAM J. Sci. Comput. **35**(2), 957–986 (2013). https://doi.org/10.1137/120885899
71. Pazner, W.: Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting. Comput. Methods Appl. Mech. Eng. **382**, 113876 (2021). https://doi.org/10.1016/j.cma.2021.113876
72. Persson, P.-O., Peraire, J.: Sub-cell shock capturing for discontinuous Galerkin methods. In: 44th AIAA Aerospace Sciences Meeting and Exhibit, AIAA 2006-112. AIAA (2006). https://doi.org/10.2514/6.2006-112
73. Rackauckas, C., Nie, Q.: DifferentialEquations.jl-A performant and feature-rich ecosystem for solving differential equations in Julia. J. Open Res. Softw. **5**(1), 15 (2017). https://doi.org/10.5334/jors.151
74. Ranocha, H., Dalcin, L., Parsani, M., Ketcheson, D.I.: Optimized Runge-Kutta methods with automatic step size control for compressible computational fluid dynamics. Commun. Appl. Math. Comput. **4**, 1191–1228 (2021). https://doi.org/10.1007/s42967-021-00159-w
75. Ranocha, H., Gassner, G.J.: Preventing pressure oscillations does not fix local linear stability issues of entropy-based split-form high-order schemes. Commun. Appl. Math. Comput. **4**(3), 880–903 (2021). https://doi.org/10.1007/s42967-021-00148-z
76. Ranocha, H., Glaubitz, J., Öffner, P., Sonar, T.: Stability of artificial dissipation and modal filtering for flux reconstruction schemes using summation-by-parts operators. Appl. Numer. Math. **128**, 1–23 (2018). https://doi.org/10.1016/j.apnum.2018.01.019
77. Ranocha, H., Schlottke-Lakemper, M., Chan, J., Rueda-Ramirez, A.M., Winters, A.R., Hindenlang, F., Gassner, G.J.: Efficient implementation of modern entropy stable and kinetic energy preserving discontinuous Galerkin methods for conservation laws (2021). arXiv: 2112.10517
78. Ranocha, H., Schlottke-Lakemper, M., Winters, A.R., Faulhaber, E., Chan, J., Gassner, G.: Adaptive numerical simulations with Trixijl: a case study of Julia for scientific computing. JuliaCon Proceedings 1(1), 77 (2022). https://doi.org/10.21105/jcon.00077
79. Ranocha, H., Winters, A.R., Castro, H.G., Dalcin, L., Schlottke-Lakemper, M., Gassner, G.J., Parsani, M.: Reproducibility repository for “on error-based step size control for discontinuous Galerkin methods for compressible fluid dynamics”. https://github.com/trixi-framework/paper-2022-stepsize_control (2022). https://doi.org/10.5281/zenodo.7078946
80. Ranocha, H.: Generalised summation-by-parts operators and entropy stability of numerical methods for hyperbolic balance laws. PhD thesis, TU Braunschweig, Braunschweig (February 2018)
81. Ranocha, H.: Entropy conserving and kinetic energy preserving numerical methods for the Euler equations using summation-by-parts operators. In: Sherwin, S.J., Moxey, D., Peiró, J., Vincent, P.E., Schwab, C. (eds.) Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2018. Lecture Notes in Computational Science and Engineering, vol. 134, pp. 525–535. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39647-3_42
82. Rivers, M.B.: NASA common research model: a history and future plans. AIAA SciTech. **1**, 1–36 (2019). https://doi.org/10.2514/6.2019-3725
83. Rojas, D., Boukharfane, R., Dalcin, L., Fernández, D.C.D.R., Ranocha, H., Keyes, D.E., Parsani, M.: On the robustness and performance of entropy stable discontinuous collocation methods. J. Comput. Phys. **426**, 109891 (2021). https://doi.org/10.1016/j.jcp.2020.109891
84. Saad, T., Karam, M.: Stable timestep formulas for high-order advection-diffusion and Navier-Stokes solvers. Comput. Fluids **244**, 105564 (2022). https://doi.org/10.1016/j.compfluid.2022.105564
85. Schlottke-Lakemper, M., Winters, A.R., Ranocha, H., Gassner, G.J.: A purely hyperbolic discontinuous Galerkin approach for self-gravitating gas dynamics. J. Comput. Phys. **442**, 110467 (2021). https://doi.org/10.1016/j.jcp.2021.110467
86. Sjögreen, B., Yee, H.: High order entropy conservative central schemes for wide ranges of compressible gas dynamics and MHD flows. J. Comput. Phys. **364**, 153–185 (2018). https://doi.org/10.1016/j.jcp.2018.02.003

87. Sonntag, M., Munz, C.-D.: Shock capturing for discontinuous Galerkin methods using finite volume subcells. In: Fuhrmann, J., Ohlberger, M., Rohde, C. (eds.) Finite Volumes for Complex Applications VII-Elliptic, Parabolic and Hyperbolic Problems. Springer Proceedings in Mathematics & Statistics, vol. 78, pp. 945–953. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05591-6_96
88. Söderlind, G.: Automatic control and adaptive time-stepping. Numer. Algorithms **31**(1/2/3/4), 281–310 (2002). https://doi.org/10.1023/A:1021160023092
89. Söderlind, G.: Digital filters in adaptive time-stepping. ACM Trans. Math. Softw. (TOMS) **29**(1), 1–26 (2003). https://doi.org/10.1145/641876.641877
90. Söderlind, G.: Time-step selection algorithms: adaptivity, control, and signal processing. Appl. Numer. Math. **56**(3/4), 488–502 (2006). https://doi.org/10.1016/j.apnum.2005.04.026
91. Söderlind, G., Wang, L.: Adaptive time-stepping and computational stability. J. Comput. Appl. Math. **185**(2), 225–243 (2006). https://doi.org/10.1016/j.cam.2005.03.008
92. Uh Zapata, M., Gamboa Salazar, L., Itzá Balam, R., Nguyen, K.D.: An unstructured finite-volume semi-coupled projection model for bed load sediment transport in shallow-water flows. J. Hydraulic Res. **59**(4), 545–558 (2021). https://doi.org/10.1080/00221686.2020.1786740
93. Vincent, P., Witherden, F., Vermeire, B., Park, J.S., Iyer, A.: Towards green aviation with Python at petascale. In: SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, pp. 1–11 (2016). https://doi.org/10.1109/SC.2016.1
94. Ware, J., Berzins, M.: Adaptive finite volume methods for time-dependent PDEs. In: Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations, pp. 417–430. Springer, New York (1995)
95. Wintermeyer, N., Winters, A.R., Gassner, G.J., Kopriva, D.A.: An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry. J. Comput. Phys. **340**, 200–242 (2017). https://doi.org/10.1016/j.jcp.2017.03.036
96. Winters, A.R., Gassner, G.J.: A comparison of two entropy stable discontinuous Galerkin spectral element approximations for the shallow water equations with non-constant topography. J. Comput. Phys. **301**, 357–376 (2015). https://doi.org/10.1016/j.jcp.2015.08.034
97. Woodward, P., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. J. Comput. Phys. **54**(1), 115–173 (1984). https://doi.org/10.1016/0021-9991(84)90142-6
98. Yan, G., Kaur, S., Banks, J.W., Hicken, J.E.: Entropy-stable discontinuous Galerkin difference methods for hyperbolic conservation laws. J. Comput. Appl. Math. **422**, 114885 (2023)
99. Zhang, X., Shu, C.-W.: Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments. Proc. R. Soc. London A: Math. Phys. Eng. Sci. **467**(2134), 2752–2776 (2011). https://doi.org/10.1098/rspa.2011.0153

Authors and Affiliations

Hendrik Ranocha[1] · Andrew R. Winters[2] · Hugo Guillermo Castro[3] · Lisandro Dalcin[3] · Michael Schlottke-Lakemper[4,5] · Gregor J. Gassner[6,7] · Matteo Parsani[3,8]

Andrew R. Winters
andrew.ross.winters@liu.se

Hugo Guillermo Castro
castrohgui@gmail.com

Lisandro Dalcin
dalcinl@gmail.com

Michael Schlottke-Lakemper
m.schlottke-lakemper@acom.rwth-aachen.de

Gregor J. Gassner
ggassner@uni-koeln.de

Matteo Parsani
matteo.parsani@kaust.edu.sa

[1] Applied Mathematics, University of Hamburg, Bundesstr. 55, 20146 Hamburg, Germany

[2] Department of Mathematics; Applied Mathematics, Linköping University, Linköping, Sweden

[3] Extreme Computing Research Center (ECRC), Computer Electrical and Mathematical Science and Engineering Division (CEMSE), King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia

[4] Applied and Computational Mathematics, RWTH Aachen University, Aachen, Germany

[5] High-Performance Computing Center Stuttgart, University of Stuttgart, Stuttgart, Germany

[6] Department of Mathematics and Computer Science, University of Cologne, Cologne, Germany

[7] Center for Data and Simulation Science, University of Cologne, Cologne, Germany

[8] Physical Science and Engineering Division (PSE), King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia