Performance of high-order Godunov-type methods in simulations of astrophysical low Mach number flows

G. Leidi¹, R. Andrassy¹, W. Barsukow², J. Hig¹,³, P. V. F. Edelmann⁴, and F. K. Röpke¹,⁵

¹ Heidelberger Institut für Theoretische Studien, Schloss-Wolfsbrunnenweg 35, D-69118 Heidelberg, Germany
e-mail: giovanni.leidi@ihs-its.org
² Bordeaux Institute of Mathematics, Bordeaux University and CNRS/UMR5251, Talence, 33405 France
³ High-Performance Computing Center Stuttgart, Nobelstraße 19, 70569 Stuttgart, Germany
⁴ Computer, Computational and Statistical Sciences (CCS) Division and Center for Theoretical Astrophysics (CTA), Los Alamos National Laboratory, Los Alamos, PO Box 1663, NM 87545, USA
⁵ Zentrum für Astronomie der Universität Heidelberg, Institut für Theoretische Astrophysik, Philosophenweg 12, D-69120 Heidelberg, Germany

Received 8 December 2023 / Accepted 23 February 2024

ABSTRACT

High-order Godunov methods for gas dynamics have become a standard tool for simulating different classes of astrophysical flows. Their accuracy is mostly determined by the spatial interpolant used to reconstruct the pair of Riemann states at cell interfaces and by the Riemann solver that computes the interface fluxes. In most Godunov-type methods, these two steps can be treated independently, so that many different schemes can in principle be built from the same numerical framework. Because astrophysical simulations often test out the limits of what is feasible with the computational resources available, it is essential to find the scheme that produces the numerical solution with the desired accuracy at the lowest computational cost. However, establishing the best combination of numerical options in a Godunov-type method to be used for simulating a complex hydrodynamic problem is a nontrivial task. In fact, formally more accurate schemes do not always outperform simpler and more diffusive methods, especially if sharp gradients are present in the flow. For this work, we used our fully compressible SEVEN-LEAGUE Hygro (SLH) code to test the accuracy of six reconstruction methods and three approximate Riemann solvers on two- and three-dimensional (2D and 3D) problems involving subsonic flows only. We considered Mach numbers in the range from 10⁻³ to 10⁻¹, which are characteristic of many stellar and geophysical flows. In particular, we considered a well-posed, 2D, Kelvin–Helmholtz instability problem and a 3D turbulent convection zone that excites internal gravity waves in an overlying stable layer. Although the different combinations of numerical methods converge to the same solution with increasing grid resolution for most of the quantities analyzed here, we find that (i) there is a spread of almost four orders of magnitude in computational cost per fixed accuracy between the methods tested in this study, with the most performant method being a combination of a low-dissipation Riemann solver and a sextic reconstruction scheme; (ii) the low-dissipation solver always outperforms conventional Riemann solvers on a fixed grid when the reconstruction scheme is kept the same; (iii) in simulations of turbulent flows, increasing the order of spatial reconstruction reduces the characteristic dissipation length scale achieved on a given grid even if the overall scheme is only second order accurate; (iv) reconstruction methods based on slope-limiting techniques tend to generate artificial, high-frequency acoustic waves during the evolution of the flow; and (v) unlimited reconstruction methods introduce oscillations in the thermal stratification near the convective boundary, where the entropy gradient is steep.

Key words. Convection – Hydrodynamics – Instabilities – Methods: numerical – Turbulence – Waves

1 1. Introduction

High-resolution schemes for gas dynamics (see, e.g., van Leer 1979; Colella & Woodward 1984; Harten et al. 1987; Colella 1990; Liu et al. 1994; Jiang & Shu 1996; Colella & Sekora 2008; Toro 2009; Balsara 2017) are routinely used for modeling a broad variety of astrophysical flow phenomena. Their popularity derives from their conservation properties and robustness, which allow them to accurately capture both smooth and discontinuous solutions on the same computational grid without sacrificing numerical stability.

These schemes are based on higher-order extensions of the original first-order accurate method of Godunov (1959) and their time-integration algorithm is typically carried out in three steps. First, a pair of Riemann states is reconstructed at each grid cell interface by applying high-order monotonic interpolants to a set of cell-averaged hydrodynamic quantities. Second, the resulting Riemann problems are solved (either exactly or approximately) to obtain fluxes across every cell boundary. Finally, the cell surface integral of the fluxes is evaluated, allowing the cell-volume-averaged state quantities to be advanced in time.

In most high-order Godunov schemes, the solution strategy of the Riemann problem is independent of the spatial interpolant used for reconstructing the Riemann states. Therefore, many different schemes can be built from the same numerical framework. The choices made in the construction of a particular scheme, however, do have a strong effect on its accuracy, that is the dif-
ference between the numerical \((U)\) and the true \((u)\) solution,
\[
\|U - u\| = O(\Delta x^n) + O(\Delta t^m),
\]
computed in some norm \(\| \cdot \|\) (see, e.g., LeVeque 2002). Here, \(\Delta x\)
is the width of the grid cell and \(\Delta t\) is the time step. Although the
formal order of the spatial and temporal accuracy of a Godunov-
like scheme, \("m"\) and \("n"\) in Eq. (1), can be derived for smooth
flows, they do not give any information about the magnitude of
the numerical errors generated on a given grid, which is problem-
dependent. The convergence rates can also be significantly lower
than the formal order of accuracy of the scheme for problems
that admit nonsmooth solutions. Consequently, formally higher-
than-second-order interpolants do not always outperform sim-
pler linear spatial reconstruction schemes when large gradients
or discontinuities are present in the flow (Greenough & Rider
2003). Moreover, if the flow is stochastic or chaotic, as in the
case of turbulence, convergence may not be achieved in the sense
of Eq. (1), but rather the quality of the numerical results can only
be judged in terms of the global or ensemble-averaged quantities
that characterize the flow and its evolution.

Given these considerations, it is impossible to generalize the
convergence properties of a certain combination of numerical
methods in a Godunov-type scheme, so they have to be ex-
plored by running numerical tests. For such tests to be signifi-
cant, they have to be challenging enough and close to the actual
application case. The performance of the numerical scheme is
another crucial aspect to be considered alongside its accuracy,
especially in astrophysical simulations, which often test out the
limits of what is feasible nowadays with available computational
resources. Therefore, the question arises of what combination of
different ingredients in a Godunov-type scheme should be used
to produce the desired solution at a minimal computational cost.

Several comparison studies have been presented in the lit-

erature with the aim of shedding light on the behavior of dif-
ferent high-resolution schemes in simulations of complex hy-
drodynamic phenomena, such as forced turbulence (Klingenberg
et al. 2007; Kritsuk et al. 2011; Sun & Kara 2015; Radice et al.
2015; See & Ryu 2023), convection (Müller 2020), jet evolu-
tion (Beckwith & Stone 2011; Musoke et al. 2020), magneto-
rotation, and instabilities in accretion disks (Flock et al. 2010),
Richtmyer-Meshkov instabilities (Latini et al. 2007), and shear
instabilities (McNally et al. 2012; Lecoanet et al. 2017). These
numerical experiments focused on supersonic or mildly subsonic
flow regimes, for which Godunov-type methods are highly opti-
nized (see, e.g., LeVeque 2002; Toro 2009). Nonetheless, high-
resolution schemes have also been proven to be a powerful tool
for modeling regimes of low Mach numbers \((M \approx |V|/c \lesssim 0.1,\)
where \(V\) is the fluid velocity and \(c\) is the sound speed), especially
in simulations of terrestrial (see, e.g., Day & Bell 2000; Klein
2009; Dumbser et al. 2009; Motheau et al. 2018) and stellar
(stee, e.g., Meakin & Arnett 2007; Muthsam et al. 2010; Wood-
ward et al. 2014; Goffrey et al. 2017; Müller 2020; Horst et al.
2021; Canivez Cussia & Teysier 2022) flows. To our knowl-
edge, other than idealized tests, no extensive work along the lines
of the aforementioned comparison studies has been presented for
low-Mach-number flows yet. Only a few studies evaluated the
impact of the order of the spatial and temporal discretization in
the numerical scheme on the properties of highly subsonic turbu-
ent flows, but they kept the Riemann solver fixed (Wongwathan-

In this work, we used our fully compressible SEVEN-LEAGUE
Hydro (SLH) code to test 18 different combinations of spatial re-
construction schemes and Riemann solvers on two test problems
in which flows are highly subsonic \((10^{-3} \lesssim M \lesssim 10^{-1})\). In partic-
ular, we considered a two-dimensional (2D) Kelvin–Helmholtz
instability with smooth initial conditions and a 3D, turbulent convection zone that entrains material from an upper, stably
stratified layer, where internal waves are free to propagate. The
initial conditions of the latter test were adopted from the work of
Andrassy et al. (2022). Here, we opted to reduce the strength of
the heat source driving the convection in order to achieve lower
convective speeds than those obtained by Andrassy et al. (2022).
Also, contrary to that work, we provided performance measure-
ments for all the methods tested in our study. In both tests, we
conducted a resolution study to analyze the convergence of the
numerical results obtained by each method.

The paper is structured as follows: in Sect. 2, we provide a
detailed description of the equations solved and the numeri-
ical methods included in this study. In Sect. 3, we measure the
convergence properties of different Godunov-type schemes for
the 2D, Kelvin–Helmholtz instability test problem (see Sect. 3.1)
and for the 3D setup involving turbulent convection, convective
boundary mixing, and wave excitation (see Sect. 3.2). In Sect. 4,
we use the kinetic energy spectrum of the convective flows sim-
ulated in the latter test, which is close to a real astrophysical
application, to provide measurements of the computational cost
per fixed accuracy for each method. Finally, in Sect. 5, we sum-
marize the main results and we give some guidance on which
methods to use for specific applications.

2. Methods

2.1. Governing equations

We solved the fully compressible, inviscid Euler equations with
a source term \(S\) in the integral form
\[
\frac{1}{|\Omega|} \int_{\Omega} \frac{\partial}{\partial t} \left( \int_{\Omega} U \, d\Omega \right) + \frac{1}{|\Omega|} \int_{\partial\Omega} T \cdot n \, dA = \frac{1}{|\Omega|} \int_{\partial\Omega} S \, dA,
\]
where
\[
U = \begin{bmatrix}
\rho \\
p \\
\rho u \\
\rho v \\
\rho w \\
\rho c_{\text{tot}} \\
\rho X
\end{bmatrix}
\]
is the set of conserved quantities, \(|\Omega|\) is the volume of a fluid element \(\Omega\) enclosed by a surface \(\partial\Omega\), \(n\) is the outward normal vector to the surface, and \(T = [F|G|H]\) is a tensor defined by the flux vectors
\[
F = \begin{bmatrix}
\rho u \\
\rho u^2 + p \\
\rho u v \\
\rho u w (p\rho_{\text{tot}} + p) u \\
\rho u X
\end{bmatrix}, \quad G = \begin{bmatrix}
\rho v \\
\rho v u \\
\rho v^2 + p \\
\rho v w (p\rho_{\text{tot}} + p) v \\
\rho v X
\end{bmatrix}, \quad H = \begin{bmatrix}
\rho w \\
\rho w u \\
\rho w v \\
\rho w w (p\rho_{\text{tot}} + p) w \\
\rho w X
\end{bmatrix}
\]
\[
(4)
\]
Here, \(\rho\) denotes the mass density, \(V = (u, v, w)\) the velocity field,
\(\epsilon_{\text{int}} = \frac{1}{2}|V|^2\) the total energy per unit mass, \(\epsilon_{\text{int}}\) the specific
internal energy, and \(X\) the mass fraction of a passive scalar used
as a tracer advected with the fluid. The system represented by
Eq. (2) is closed by an equation of state (EoS), which gives the
gas pressure as a function of the density and internal energy,
\[
p = p(\rho, \epsilon_{\text{int}}).
\]
(5)
In this work, we only considered a perfect gas with a given adiabatic index $\gamma$, for which

\[ p(\rho, e_{\text{int}}) = (\gamma - 1)\rho e_{\text{int}}. \quad (6) \]

For modeling the test problem introduced in Sect. 3.2, which involves the presence of a gravitational field, $g = (g_x, g_y, g_z)$, the corresponding source term,

\[ S_{\text{grav}} = \begin{bmatrix} 0 \\ \rho g_x \\ \rho g_y \\ \rho g_z \\ \rho g \cdot V \end{bmatrix}, \quad (7) \]

must be included in the right-hand-side term of Eq. (2). Because we assumed $g$ to be time-independent, we opted to solve an equivalent form of Eq. (2), in which the gravitational potential $\phi$ was directly added to $e_{\text{int}}$, eliminating the $\rho g \cdot V$ source term from the energy equation,

\[ e_{\text{tot}} \mapsto e_{\text{tot}} + \phi, \quad S_{\text{grav}} = \begin{bmatrix} 0 \\ \rho g_x \\ \rho g_y \\ \rho g_z \\ \rho g \cdot V \end{bmatrix} \mapsto \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (8) \]

Numerical schemes that solve this form of Eq. (2) are capable of conserving the total energy of the system over time exactly. Such a property is particularly important in low-Mach-number hydrodynamics, where even small energy conservation errors can become comparable to the kinetic energy content of the flows (Müller 2020; Edelmann et al. 2021).

### 2.2. Seven-League Hydro code

In our study, Eq. (2) was solved numerically using the Seven-League Hydro code (SLH, Miczek 2013; Edelmann 2014), which was originally developed to model the broad variety of hydrodynamic processes that characterize the deep interiors of stars, such as shear instabilities (Edelmann et al. 2017), excitation of internal waves (Horst et al. 2020), convective boundary mixing (Horst et al. 2021; Andrassy et al. 2022, 2023), and turbulent dynamos (Leidi et al. 2022, 2023). SLH makes use of the finite-volume discretization on an arbitrarily curvilinear, but logically rectangular, Eulerian grid to retain the conservation properties of the fluid-dynamics equations. Numerical solutions to Eq. (2) are computed by means of Godunov-type methods based on the definition of Riemann problems at cell interfaces. The code is parallelized using the Message Passing Interface (MPI) and it has been proven to scale up to several hundred thousand processes (Edelmann & Röpke 2016).

SLH allows the user to choose among many different numerical options at compile time, which makes this code perfectly suited to run the comparison study introduced in Sect. 1. In particular, in addition to the well-known Rusanov (Rusanov 1962), Roe (Roe 1981), and Harten-Lax-van Leer-Contact (HLLC, Toro et al. 1994) approximate Riemann solvers, SLH adopts special low-Mach-number methods (Liou 2006; Miczek et al. 2015; Minoshima & Miyoshi 2021) to reduce the excessive numerical dissipation introduced by shock-capturing schemes at low Mach numbers (see Sect. 2.4). A wide spectrum of spatial reconstruction methods can be used to generate a pair or Riemann states at each grid cell interface, ranging from (first-order accurate) constant reconstruction to very high order methods, some of which are described in Sect. 2.3. In problems involving the presence of a gravitational field, the deviation well-balancing method (Berberich et al. 2021; Edelmann et al. 2021) is used to preserve hydrostatic solutions and to reduce the strength of spurious flows generated by grid discretization errors in strongly stratified media.

In SLH, the cell-volume-averaged source term and the cell-surface-averaged fluxes are approximated using the midpoint method, making the code at best second-order accurate in space. On a 3D, evenly spaced Cartesian grid, the final expressions for these integrals read

\[ \frac{1}{\Delta V} \int_{\Omega_{i,j,k}} S \, dx \, dy \, dz = S_{i,j,k} + O(\Delta x^2), \quad (9) \]

\[ \frac{1}{\Delta A} \int_{\Omega_{i,j,k}} F \, dy \, dz = F_{i+1/2,j,k} + O(\Delta x^2), \quad (10) \]

\[ \frac{1}{\Delta A} \int_{\Omega_{i,j,k}} G \, dx \, dz = G_{i,j+1/2,k} + O(\Delta x^2), \quad (11) \]

\[ \frac{1}{\Delta A} \int_{\Omega_{i,j,k}} H \, dx \, dy = H_{i,j+1/2,k} + O(\Delta x^2), \quad (12) \]

where $S_{i,j,k}$ is the point value of the source term $S$ at the center of the cell represented by the set of indices $(i, j, k)$ and vector quantities such as $F_{i+1/2,j,k}$ refer to the face-centered value of the flux at the boundary between two adjacent cells, in this case $(i, j, k)$ and $(i + 1, j, k)$. The volume of the cell and the surface of a cell face are $\Delta V = \Delta x^3$ and $\Delta A = \Delta x^2$, respectively. The discretized source term and fluxes in Eqs. (9)–(12) are used to build a semi-discrete version of Eq. (2) which leaves the problem continuous in time, following the method of lines (see, e.g., LeVeque 2002),

\[ \frac{\partial \mathbf{U}_{i,j,k}}{\partial t} = -\frac{1}{\Delta x} \left( F_{i+1/2,j,k} - F_{i-1/2,j,k} + G_{i,j+1/2,k} - G_{i,j-1/2,k} + H_{i,j,k+1/2} - H_{i,j,k-1/2} \right) + S_{i,j,k}. \quad (13) \]

The time update on the cell-volume-averaged conserved variables, $\mathbf{U}_{i,j,k}$, is then carried out in a dimensionally unsplit fashion using explicit or implicit time stepping. Here, we only considered a limited set of all of the numerical methods available in SLH to avoid constructing a too large parameter space. In the following sections, we provide a detailed description of the algorithms that we used for running the tests presented in Sect. 3.1 and 3.2.

### 2.3. Spatial reconstruction methods

We used six reconstruction methods as summarized in Table 1. We did not attempt to be exhaustive in the choice of our methods, which is why, for example, essentially nonoscillatory (ENO) and weighted ENO (WENO) schemes (see, e.g., Liu et al. 1994; Jiang & Shu 1996; Shu 2009) were left out from our comparison study.\footnote{The influence of the parameter that occurs in the smoothness indicators of the compact third order WENO scheme of Kolb (2014) on the generation of sound waves by turbulent flows is discussed in Appendix C.} We also excluded multidimensional reconstruction...
methods. However, the methods we did include cover a wide range in complexity and formal order of accuracy and several of them are often used in stellar hydrodynamics.

The simplest are unlimited linear (LIN) and parabolic (PAR) methods, which have proved to be well-behaved in implicit simulations of slow flows using our SLH code (e.g., Horst et al. 2020, 2021; Andrassy et al. 2023). Their main disadvantage – the generation of artificial oscillations around steep gradients – can be eliminated using slope limiters. Out of a wide spectrum of limiters available, we have decided to include the popular van Leer limiter (van Leer 1974) in combination with linear reconstruction (LIN+VL). This limiter has the total-variation-diminishing (TVD) property and eliminates the oscillations completely.3 As examples of higher-order methods with limiters, we included two versions of the widely used piecewise-parabolic method (PPM): Colella & Woodward (1984, CW84 hereinafter) and Colella & Sekora (2008, CS08 hereinafter), for which we introduce the acronyms PPM84 and PPM08, respectively. We also constructed a hybrid method, which we named piecewise sextic hybrid (PSH). It combines unlimited sextic reconstruction for dynamic variables (ρ, V, p) with the PPM08 method for passive scalars. We provide lower-order alternatives to the PSH method in Appendix A, although we did not test them in this study.

In the remainder of this section, we provide detailed descriptions of all the six reconstruction methods in unified notation. We do this to (i) maximize the reproducibility of our work, (ii) simplify the methods’ more general original forms (e.g., for nonuniform grids), (iii) make clear what choices we made if several options were available, and (iv) to point out a few typographic mistakes in the original works.

We refer to the variable being reconstructed as a in Sections 2.3.1–2.3.6. Although reconstruction can be performed in physical coordinates, we chose to do it in logical coordinates defined by the cell index i, which leads to much simpler expressions. This choice did not influence our results in any way since we used uniform Cartesian grids. In some cases, we introduce the continuous linear logical coordinate ζ such that the coordinate of the center of cell i is ζi = i and the left and right cell interfaces are located at ζi−1/2 = i − 1/2 and ζi+1/2 = i + 1/2, respectively. The reconstruction is usually discontinuous at the interfaces. We use the notation ai+1/2,1 and ai+1/2,2 to denote the reconstructed states on the left and right side of the interface at ζi+1/2, respectively. Reconstruction is performed along each spatial axis separately, that is to say the reconstruction procedure is always one-dimensional. To highlight the difference between cell averages and point values, we use the notation ai for the average value of a within cell i.

In this work, we always reconstructed the set of cell-volume-averaged primitive variables 〈q〉, where q = (ρ, V, p, X), because it helps reducing oscillations near discontinuities as compared to reconstructing cell-volume-averaged conserved quantities 〈U〉.

In SLH, transformations between 〈q〉 and 〈U〉 are performed using 2nd-order approximations,

\[
\begin{align*}
\bar{q} &= \xi(U) + O(\Delta x^2), \\
\bar{U} &= \xi^{-1}(\bar{q}) + O(\Delta x^2),
\end{align*}
\] (14)

where ξ is a nonlinear, invertible transformation,

\[
\xi(U) = \begin{bmatrix}
U_1 \\
U_2/U_1 \\
U_3/U_1 \\
U_4/U_1 \\
(\gamma - 1) \left( U_5 - \frac{1}{2U_1} \left( U_2^2 + U_3^2 + U_4^2 \right) \right) \\
U_6/U_1
\end{bmatrix},
\] (15)

and Ui is the i-th component of U.

2.3.1. The LIN method

The unlimited linear reconstruction method is based on a linear approximation to the underlying function a(ζ). Its slope δi (in cell-index coordinates) is estimated using the central difference (the Fromm method)

\[
\delta_i = \frac{\overline{a}_{i+1} - \overline{a}_{i-1}}{2} = \Delta x \partial_\xi a_{i\#} + O(\Delta x^3),
\] (16)

using which we obtain the reconstructed states

\[
a_{i-1/2,R} = \overline{a}_i + \frac{\delta_i}{2} + O(\Delta x^2),
\] (17)

\[
a_{i+1/2,L} = \overline{a}_i + \frac{\delta_i}{2} + O(\Delta x^2),
\] (18)

The LIN method is exact wherever a(ζ) is locally linear, 2nd-order accurate for general but smooth functions a(ζ), and it requires two ghost cells at domain boundaries.4

2.3.2. The LIN+VL method

The van-Leer-limited linear reconstruction method is particularly easy to describe in terms of the 2nd-order-accurate, interface-centered slopes

\[
\delta_{i-1/2} = \overline{a}_i - \overline{a}_{i-1},
\] (19)

\[
\delta_{i+1/2} = \overline{a}_i - \overline{a}_{i+1},
\] (20)

The final slope δlim is then obtained by applying the limiter of van Leer (1974),

\[
\delta_{i,\text{lim}} = \begin{cases} 2\delta_{i+1/2} \overline{a}_{i+1/2} - 3\delta_{i-1/2} + 2\delta_{i-1/2} & \text{if } \delta_{i-1/2} > 0 \\ 0 & \text{otherwise,} \end{cases}
\] (21)

which gives the reconstructed states

\[
a_{i-1/2,R} = \overline{a}_i + \frac{\delta_{i,\text{lim}}}{2} + O(\Delta x^2),
\] (22)

\[
a_{i+1/2,L} = \overline{a}_i + \frac{\delta_{i,\text{lim}}}{2} + O(\Delta x^2),
\] (23)

The limiter makes the slope less steep where a(ζ) is strongly curved and flat at local extrema. With a smooth and monotonic function a(ζ), the effect of the limiter weakens upon grid refinement and the left and right states converge to those provided by the LIN method. This makes the LIN+VL method formally 2nd-order accurate away from any extremum. Two ghost cells are required at domain boundaries.

4 Although the method only uses the cells i − 1, i, and i + 1, reconstruction must also be performed in the first ghost cell to fully define the Riemann problem at the domain boundary.
The resulting expressions are obtained by evaluating Eq. (24) at another limiter in PPM84 (see below), this modification is overshoots may appear within the cell. To prevent this, a second limiting step is introduced. We express it in terms of the differences

\[ a_i^- = a_{i-1/2,R} - \bar{a}_i, \quad a_i^+ = a_{i+1/2,L} - \bar{a}_i. \]  

The limiter is then defined by the variable replacements

\[ a_i^- \mapsto 0, \quad a_i^+ \mapsto 0 \quad \text{if} \quad a_i^- a_i^+ \geq 0, \]

\[ a_i^- \mapsto -2a_i^+ \quad \text{if} \quad |a_i^-| > 2|a_i^+|, \]

\[ a_i^+ \mapsto -2a_i^- \quad \text{if} \quad |a_i^+| > 2|a_i^-|. \]

The reconstructed states \( a_{i-1/2,R} \) and \( a_{i+1/2,L} \) are recovered using Eqs. (29) and (30). Equation (31) is equivalent to Eq. 1.10 of CW84. This second limiter introduces discontinuities at cell interfaces in regions where the gradient of \( a(\xi) \) changes rapidly and it flattens the assumed parabola at local extrema.

Unlike CW84, we did not use the parabolic model of \( a(\xi) \) inside the cell in any way. We only needed the states at the two sides of each interface to construct Riemann problems and time integration was done using a Runge-Kutta scheme (see Sect. 2.5 for details). However, the second limiter (Eq. (31)), which is based on the parabolic model, was still needed to remove oscillations and to introduce dissipation where necessary.

Although the interpolation formula that PPM84 starts with is 4th-order accurate, the slope flattening introduced at all extrema reduces the practically attainable order of accuracy substantially for nonmonotonic solutions even if they are smooth. Our 1D experiment in Sect. B.1 gives the order of 2.3 whereas Colella & Sekora (2008) reach the order of 2.6 in a similar advection experiment with a Gaussian-shaped profile and PPM84 reconstruction. The PPM84 method requires three ghost cells at domain boundaries.

### 2.3.5. The PPM08 method

The piecewise parabolic method of CS08 is based on ideas similar to those of CW84 in constructing the PPM84 scheme and PPM08 also contains two limiters. However, the limiters are

\[ \delta_{i,\text{lim}} = \begin{cases} \text{sgn}(\delta_i) \min(|\delta_i|, 2|\delta_{i-1/2,L}|, 2|\delta_{i+1/2,L}|) & \text{if } \delta_{i-1/2} \delta_{i+1/2} > 0, \\ 0 & \text{otherwise}, \end{cases} \]  

which is the monotonized central limiter of van Leer (1977). In our implementation, we did not set \( \delta_{i,\text{lim}} = 0 \) if \( \delta_{i-1/2} \delta_{i+1/2} \leq 0 \) (i.e., at local extrema). We have tested that, thanks to the presence of another limiter in PPM84 (see below), this modification has essentially no influence on the results. However, it makes the code faster because it removes three conditional expressions per reconstruction step (we need \( \delta_{i-1/2,\text{lim}}, \delta_{i,\text{lim}}, \) and \( \delta_{i+1/2,\text{lim}} \) to obtain \( a_{i-1/2} \) and \( a_{i+1/2} \)).

### Table 1. Overview of spatial reconstruction methods used in this work.

<table>
<thead>
<tr>
<th>ID</th>
<th>Theoretical order</th>
<th>Practical order</th>
<th>Number of ghost cells</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIN</td>
<td>2</td>
<td>2.0</td>
<td>2</td>
<td>piecewise linear, no limiter</td>
</tr>
<tr>
<td>LIN+VL</td>
<td>2</td>
<td>2.0</td>
<td>2</td>
<td>piecewise linear, van Leer limiter</td>
</tr>
<tr>
<td>PAR</td>
<td>3</td>
<td>3.0</td>
<td>2</td>
<td>piecewise parabolic, no limiter</td>
</tr>
<tr>
<td>PPM84</td>
<td>4</td>
<td>2.3, 1.6</td>
<td>3</td>
<td>PPM method of Colella &amp; Woodward (1984)</td>
</tr>
<tr>
<td>PPM08</td>
<td>6</td>
<td>6.0</td>
<td>4</td>
<td>PPM method of Colella &amp; Sekora (2008)</td>
</tr>
<tr>
<td>PSH</td>
<td>7</td>
<td>7.0</td>
<td>4</td>
<td>hybrid piecewise sextic with no limiter for dynamic variables and PPM08 for passive scalars</td>
</tr>
</tbody>
</table>

Notes. ID: identifier; Theoretical order: formal order of accuracy for smooth and monotonic solutions in 1D; Practical order: asymptotic order obtained in the 1D experiments reported in Appendix B (smooth but nonmonotonic solutions). In the case of PPM84, the higher order of accuracy corresponds to linear advection and the lower one to the propagation of a sound wave using the LHLLC flux function, see Appendix B for details.
modified such that PPM08 models smooth extrema instead of flattening them.

In PPM08, the first estimate of \( a_{i+1/2} \) is obtained using the 6th-order-accurate interpolation formula (c.f. Eq. 17 of CS08)
\[
a_{i+1/2} = \frac{37}{60} (\bar{a}_i + \bar{a}_{i+1}) - \frac{2}{15} (\bar{a}_{i-1} + \bar{a}_{i+2}) + \frac{1}{60} (\bar{a}_{i-2} + \bar{a}_{i+3})
\]

\[
= a_{\text{lim}} + O(\Delta x^6).
\]

(32)

If \( a_{i+1/2} \) does not satisfy the condition (c.f. Eq. 13 of CS08)
\[
\min(\bar{a}_i, \bar{a}_{i+1}) \leq a_{i+1/2} \leq \max(\bar{a}_i, \bar{a}_{i+1})
\]

(34)
a limiter is applied. It is based on three 2nd-order-accurate second derivatives,
\[
(D^2 a)_{i+1/2} = 3 \left( \bar{a}_i - 2a_{i+1/2} + \bar{a}_{i+1} \right) = \partial^2 a_{\text{lim}} \Delta x^2 + O(\Delta x^4),
\]

(35)
\[
(D^2 a)_i = \bar{a}_i - 2\bar{a}_i + \bar{a}_{i+1} = \partial^2 a_{\text{lim}} \Delta x^2 + O(\Delta x^4),
\]

(36)
\[
(D^2 a)_{i-1} = \bar{a}_i - 2\bar{a}_i + \bar{a}_i - 2 = \partial^2 a_{\text{lim}} \Delta x^2 + O(\Delta x^4).
\]

(37)

In Eq. (35), the more common finite difference formula with a prefactor 4 would arise if the involved quantities were of the same kind, meaning all three point values, or all three averages. These derivatives are combined to obtain a limited derivative
\[
(D^2 a)_{i+1/2,\text{lim}} = \min \left( \left| (D^2 a)_{i+1/2} \right|, C \left| (D^2 a)_i \right| \right)
\]

(38)
\[
\text{if all three derivatives have the same sign and}
\]

(39)
\[
(D^2 a)_{i+1/2,\text{lim}} = 0
\]

otherwise. We used \( C = 1.25 \) in Eq. (38). The limited derivative is then used to modify the value of \( a_{i+1/2} \),
\[
a_{i+1/2} \rightarrow \frac{1}{2} (\bar{a}_i + a_{i+1}) - \frac{1}{6} (D^2 a)_{i+1/2,\text{lim}}.
\]

(40)

Equation 19 of CS08 is equivalent to our Eq. (40) except for the factor in front of the second term, which is \( \frac{1}{6} \) in CS08. This is likely a typographical error because the factor of \( \frac{1}{5} \) is needed to obtain the original, unlimited value of \( a_{i+1/2} \) when the solution is so smooth that the limiter does not change the second derivative significantly.

Just as in the PPM84 method, the interpolated value \( a_{i+1/2} \) is initially assigned to both \( a_{i+1/2,1} \) and \( a_{i+1/2,2} \), meaning there is no discontinuity at the interface. The second limiting step depends on whether cell \( i \) is in the vicinity of a local extremum or not. If
\[
(\bar{a}_i - a_{i+1/2,R}) (a_{i+1/2,L} - \bar{a}_i) \leq 0 \quad \text{or} \quad (\bar{a}_i - a_{i+1/2,L}) (a_{i+1/2,R} - \bar{a}_i) \leq 0,
\]

(41)
cell \( i \) is close to a local extremum, which should be preserved if smooth enough. The second derivatives
\[
(D^2 a)_{i} = 6(a_{i-1/2,R} - 2\bar{a}_i + a_{i+1/2,L}),
\]

(42)
\[
(D^2 a)_{i-1} = \bar{a}_i - 2\bar{a}_i + \bar{a}_i,
\]

(43)
\[
(D^2 a)_{i} = \bar{a}_{i-1} - 2\bar{a}_i + \bar{a}_{i+1},
\]

(44)
\[
(D^2 a)_{i+1} = \bar{a}_i - 2\bar{a}_i + \bar{a}_{i+2}.
\]

(45)

are combined to judge the solution’s smoothness (CS08 have one wrong index in their equivalent of our Eq. (43), c.f. their Eq. 21). We then set
\[
(D^2 a)_{i,\text{lim}} = \min \left( \left| (D^2 a)_i \right|, C \left| (D^2 a)_{i-1} \right|, C \left| (D^2 a)_{i} \right| \right)
\]

(46)

if all four second derivatives have the same sign and
\[
(D^2 a)_{i,\text{lim}} = 0
\]

(47)
otherwise. Finally, the reconstructed states are updated,
\[
a_{i-1/2,R} \rightarrow a_i + (a_{i-1/2,R} - \bar{a}_i) \frac{(D^2 a)_{i,\text{lim}}}{(D^2 a)_i},
\]

(48)
\[
a_{i+1/2,L} \rightarrow a_i + (a_{i+1/2,L} - a_i) \frac{(D^2 a)_{i,\text{lim}}}{(D^2 a)_i}.
\]

(49)

If \( |(D^2 a)_i| < \epsilon = 10^{-12} \) we do not modify the reconstructed states in this step. Equations (48) and (49) are equivalent to Eq. 23 of CS08.

If the condition in Eq. (41) is not satisfied, meaning cell \( i \) is not in the vicinity of a local extremum, we use Eqs. (29)-(31) instead of Eqs. (48) and (49) to limit the reconstructed states. CS08 propose to use a slightly less restrictive limiter away from extrema (their Eq. 26), but that limiter produces oscillations with our time-discretization scheme and we did not use it.

Just as we did in the case of PPM84, we only used the reconstructed states and not the assumed parabolic model of \( a(\zeta) \) within the cell, see Sect. 2.3.4 for details. The PPM08 method is 6th-order accurate for smooth functions \( a(\zeta) \) even if they are not monotonic. Our 1D experiment in Sect. B.1 confirms this. The PPM08 method requires four ghost cells at domain boundaries.

2.3.6. The PSH method

Whereas all of the previous reconstruction methods are applied to all variables in the same way, the PSH method is hybrid: it is an unlimited piecewise-sextic method for all dynamic variables combined with PPM08 for passive scalars. This allows us to eliminate certain issues that occur with methods containing limiters when applied to slow flows (see Sect. 3.1 for details) while essentially eliminating oscillations in the passive scalars, which could represent mass fractions. The piecewise sextic reconstruction assumes that within cell \( i \) \( a(\zeta) \) can be described by the sextic polynomial
\[
a(\zeta) = \sum_{n=0}^{6} c_n (\zeta - \zeta_i)^n.
\]

(50)

The seven coefficients \( c_n \) are uniquely determined by the requirement that the averages of \( a(\zeta) \) in cells \( i - n \) equal \( \bar{a}_{i-n} \) for \( n = 0, 1, \ldots, 6 \). The reconstructed states are then obtained by evaluating Eq. (50) at \( \zeta_{i-1/2} \) and \( \zeta_{i+1/2} \), respectively. The result-
ing expressions are
\[
\begin{align*}
a_{i-1/2, R} &= \frac{1}{420} \left( 47 \bar{a}_{i-3} - 38 \bar{a}_{i-2} + 214 \bar{a}_{i-1} + 319 \bar{a}_i \\
&- 101 \bar{a}_{i+1} + 25 \bar{a}_{i+2} - 3 \bar{a}_{i+3} \right) + O(\Delta t^2), \\
a_{i+1/2, L} &= \frac{1}{420} \left( -3 \bar{a}_{i-3} + 25 \bar{a}_{i-2} - 101 \bar{a}_{i-1} + 319 \bar{a}_i \\
&+ 214 \bar{a}_{i+1} - 38 \bar{a}_{i+2} + 4 \bar{a}_{i+3} \right) + O(\Delta t^2),
\end{align*}
\]
(51)
(52)

The method is exact wherever \( a(\xi) \) is locally a sextic polynomial, 7th-order accurate for general but smooth functions \( a(\xi) \), and it requires four ghost cells at domain boundaries.

2.4. Approximate Riemann solvers

The reconstructed pair of primitive state quantities, \( q_{i+1/2, L, R} \), defines a Riemann problem at the cell interface \( i + 1/2 \), which SLH solves by means of 1D approximate Riemann solvers to obtain the face-centered value of the fluxes \( F_{i+1/2} \). We ran the tests presented in Sect. 3 using two widely popular flux functions, namely the RUSANOV and HLLC solvers. Because in this work we only focused on simulations of subsonic flows, for comparison we also built a low-dissipation version of HLLC following the approach of Minoshima & Miyoshi (2021), who modified the Harten-Lax-van Leer-Discontinuities (HLLD, Miyoshi & Kusano 2005) scheme for magnetohydrodynamics to diminish the magnitude of the numerical dissipation for low-Mach-number flows. The authors called this low-dissipation flux “LHLDD,” so, for consistency, we refer to the low-dissipation HLLC solver as “LHLLC” throughout the text. In the rest of this section, we summarize the main aspects of each of these solvers and provide the implementation details whenever several choices can be made for specifying the value of a certain quantity that is needed to evaluate the numerical flux.

2.4.1. RUSANOV

The RUSANOV flux is computed by adding an upwind, numerical dissipative term proportional to the maximum wave speed at the cell interface. \( S_{\text{max}} \), to every component of the central flux.

The final expression for the numerical flux reads \(^6\)
\[
\mathcal{F}(U_L, U_R) = \frac{1}{2} \left[ \mathcal{F}(U_L) + \mathcal{F}(U_R) \right] - \frac{1}{2} S_{\text{max}} (U_R - U_L),
\]
(53)
where \( U_{L,R} \) are the left and right sets of conserved quantities, respectively. In SLH, \( S_{\text{max}} \) is estimated as
\[
S_{\text{max}} = \max \left( |u_L| + c_L |u_R| + c_R \right),
\]
(54)
where \( c = (\gamma p/\rho)^{1/2} \) is the sound speed. The dissipative term in Eq. (53) scales with the Mach number of the flow \( M \) and allows

\(^5\) Although SLH was already equipped with several low-Mach solvers, such as AUSM\(^+\)-up (Liou 2006) and Mieczek-Roe (Mieczek et al. 2015), we decided not to use them in this study because they all suffer from a more restrictive stability criterion on the time step than LHLLC when used in combination with explicit time steppers, whilst all of these fluxes provide very similar results in terms of accuracy.

\(^6\) For sake of clarity, here we assume that the fluxes are computed in the \( x \) direction and dropped the indices, but analogous expressions can be obtained for the \( y \) and \( z \) directions.

the scheme to achieve numerical stability by smearout any discontinuity that may arise in the vector of state quantities \( \mathbf{U} \).

The RUSANOV solver is one of the simplest schemes that can be used to approximate the fluxes at grid cell interfaces, which makes it very efficient in terms of Floating Point Operations per Second. However, it does not take into account the complex structure of the solution arising from the Riemann problem of gas dynamics (see, e.g., Toro 2009), so the states between the two outer waves in the Riemann fan are averaged out. For this reason, this flux function is particularly diffusive for transporting contact and shear waves, which lack the self-steepening property of sound waves.

2.4.2. HLLC

Different from RUSANOV, the HLLC solver restores the linearly degenerate contact and shear waves back to the structure of the solution of the Riemann problem (see Fig. 1). In this method, the numerical flux is chosen according to the sign of the wave speeds in the Riemann fan,

\[
\mathcal{F}(U_L, U_R) = \begin{cases} F_L & \text{if } 0 \leq S_L, \\
F_1^\# & \text{if } S_L < 0 \leq S^*, \\
F_R & \text{if } S^* < 0 \leq S_R, \\
F_R & \text{if } S_R < 0. \end{cases}
\]
(55)

While the computation of the physical fluxes \( F_L = \mathcal{F}(U_L) \) and \( F_R = \mathcal{F}(U_R) \) is trivial, the fluxes in the intermediate regions, \( F_1^\# \) and \( F_R^\# \), are obtained by solving the Rankine–Hugoniot jump conditions across the two outer sonic waves,

\[
\begin{align*}
S_L U_L - F_L^* &= S_L U_L - F_L, \\
S_R U_R - F_R^* &= S_R U_R - F_R.
\end{align*}
\]
(56)
(57)

Here, \( U_L^* \) and \( U_R^* \) represent the state quantities in the star regions. In order to solve Eqs. (56)–(57), proper estimates of the wave speeds \( S_L \) and \( S_R \) must be provided beforehand. SLH computes these wave speeds as

\[
\begin{align*}
S_L &= \min(a_L, a_R) - \max(c_L, c_R), \\
S_R &= \max(a_L, a_R) + \max(c_L, c_R).
\end{align*}
\]
(58)
(59)

After some assumptions and algebraic manipulations (see Toro 2009), these estimates allow Eqs. (56)–(57) to be solved for the...
where ˜(p
L
) is the selected state of the Riemann fan, 

\[
\rho v^n_L = \tilde{\alpha}_L R v_L,
\]

\[
\rho v^n_R = \tilde{\alpha}_R R v_R,
\]

and \(\tilde{\alpha}_L, \tilde{\alpha}_R\) being the speed of the intermediate wave, 

\[
S^* = \frac{\rho_L - \rho_L U_L^2}{\tilde{\alpha}_L (S_L - u_L)} - \frac{\rho_R U_R^2 - \rho_R}{\tilde{\alpha}_R (S_R - u_R)}.
\]

The gas pressure is preserved across the middle wave and takes the value 

\[
p^* = \frac{1}{2} \left( \rho_L + \rho_R + \rho_L U_L (S_L - u_L) \right) (S^* - u_L) + \rho_R (S_R - u_R) (S^* - u_R).
\]

These expressions are then inserted back into Eqs. (56)–(57) to compute \(F_L^r\) and \(F_R^r\). Finally, the interface flux is chosen according to Eq. (55).

In our study, we used a variant of the original HLLC solver of Toro et al. (1994), which allowed the low-Mach correction presented in the next section to be implemented trivially into the solver. In particular, we directly evaluated the physical fluxes in the selected state of the Riemann fan,

\[
F(U_L, U_R) = \begin{cases} 
F_L & \text{if } 0 \leq S_L, \\
F(U_L^*) & \text{if } S_L < 0 \leq S^*, \\
F(U_R^*) & \text{if } S^* < 0 \leq S_R, \\
F_R & \text{if } S_R < 0,
\end{cases}
\]

and we computed \(p^*\) using a linearized Riemann solver for the equations of gas dynamics (see, e.g., Toro 1991), 

\[
p^* = \frac{1}{2} (\rho_L + \rho_R) - \frac{1}{2} \tilde{\rho} c_h (u_R - u_L),
\]

where \(\tilde{\rho} = (c_L + c_R)/2\) and \(\tilde{\rho} = (\rho_L + \rho_R)/2\). The system in Eq. (69) is consistent with the physical fluxes in the sense that 

\[
F(U, U) = F(U)
\]

and it satisfies the Rankine–Hugoniot jump conditions across the contact wave \(S^*\) as the original solver,

\[
S^* U_L^* - F_R = S^* U_R^* - F_L.
\]

Diagnostic tests of the Kelvin–Helmholtz instability problem, described in Sect. 3.1, showed that the numerical solutions computed with our modified version and the original solver of Toro et al. (1994) are virtually indistinguishable for subsonic flows.

We stress, however, that the fluxes in Eq. (69) do not satisfy the jump conditions across the sonic waves \(S_L\) and \(S_R\) (see Eqs. (56) and (57)). Therefore, there is no guarantee that the resulting scheme preserves positivity of density and internal energy when the flow is nearly transonic, in which case effects of compressibility and non-linearities can become dominant. Such a flow regime, however, was not considered in this study.

2.4.3. LHLLC

As discussed in Sect. 2.4.2, HLLC restores the intermediate, linearly degenerate waves, so it is generally more accurate than two-wave solvers such as RUSANOV or HLL (Harten et al. 1983) in simulations involving the presence of material interfaces or the propagation of entropy waves. However, the effects of the numerical dissipation introduced by HLLC on the evolution of the flow become progressively more dominant as \(M \to 0\), thus producing unnecessarily large dissipative errors in highly subsonic velocity regimes (see, e.g., Fleischmann et al. 2020). In our variant of HLLC, this behavior is caused by the upwind term in the expression for \(p^*\) (see Eq. (70)),

\[
D(p^*) = -\frac{1}{2} \tilde{\rho} c_h (u_R - u_L).
\]

This term scales with \(M\), which is inconsistent with the scaling of pressure fluctuations dynamically generated by subsonic flows. In fact, in the asymptotic limit \(M \to 0\), the solution to the compressible Euler equations approaches the incompressible regime (Guillard & Viozat 1999), in which the gas pressure is homogeneous in space except for fluctuations proportional to \(M^2\).

At low Mach numbers, the numerical term in Eq. (73) can eventually become larger than the physical pressure fluctuation at the cell interface, thus yielding highly inaccurate numerical results.

In order to correct for the flawed scaling of the numerical dissipation introduced by HLLC-like methods, here we followed the approach described in Minoshima & Miyoshi (2021), who proposed to multiply the diffusive term in Eq. (73) by a factor \(\phi\) proportional to the local Mach number of the flow\(^7\). Such a correction was originally applied to the magnetohydrodynamic solver HLLD, but it can easily be used in HLLC by setting all magnetic field components to zero, resulting in

\[
\phi = \chi (2 - \chi),
\]

with

\[
\chi = \min \left\{ 1, \max \left( \frac{|V_L|}{c_L}, \frac{|V_R|}{c_R} \right) \right\}.
\]

The final expression for \(p^*\) then reads

\[
p^* = \frac{1}{2} (\rho_L + \rho_R) - \phi \frac{1}{2} \tilde{\rho} c_h (u_R - u_L).
\]

The resulting upwind term in this “low-dissipation” version of the HLLC flux (LHLLC) scales with \(M^2\) when the flow is subsonic, so the ratio of the numerical diffusive term to the amplitude of pressure fluctuations is independent of \(M\).

We note that the same fix cannot equally be applied to the RUSANOV flux without sacrificing numerical stability. In particular, a diffusive coefficient proportional to \(M^2\) would result in too little dissipation for sound waves. This is not the case for LHLLC, in which the complex upwinding performed in Eq. (69) guarantees that the scheme remains stable for the propagation of sound waves (see also Appendix B.1).
2.5. Time discretization

Because the acoustic Courant-Friedrichs-Lewy (CFL, Courant et al. 1928) criterion on the time step becomes excessively strict in regimes of very low Mach numbers, implicit time discretization techniques are typically better suited for simulating the evolution of such slow flows (see, e.g., Viallat et al. 2011; Miezek et al. 2015; Dumbser et al. 2019). However, we recognize that most hydrodynamic codes nowadays do not have time-implicit integration capabilities, whose implementation requires a considerable effort from code developers. Thus, to make our study easily reproducible, we decided to target in our test setups, see Sect. 3.1 and 3.2, flows with Mach numbers in the range \(10^{-3} \leq M \leq 10^{-1}\), where simple time-explicit marching schemes are still competitive with implicit ones. In this work, explicit time integration was performed in a semi-discrete fashion, in which the cell-surface integral of the fluxes and the cell-volume integral of the source terms in Eq. (2) were first separately discretized in space whilst the system was left continuous in time according to the method of lines (see Sect. 2.2). The resulting system of ordinary differential equations (see Eq. (13)), was then solved numerically to advance the cell-volume-averaged state quantities in time. To solve Eq. (13), we used the third-order accurate, strong stability preserving (SSP) RK3 method of Shu & Osher (1988), in which the update on \(\tilde{U}_{i,j}^{(n)}\) from time \(t_n\) to time \(t_{n+1} = t_n + \Delta t\) is performed in three stages,

\[
\begin{align*}
\tilde{U}_{i,j}^{(1)} &= \tilde{U}_{i,j}^{(n)} - \mathcal{R}(\tilde{U}_{i,j}^{(n)}) \Delta t, \\
\tilde{U}_{i,j}^{(2)} &= \frac{3}{4} \tilde{U}_{i,j}^{(1)} + \frac{1}{4} \tilde{U}_{i,j}^{(0)} - \frac{1}{4} \mathcal{R}(\tilde{U}_{i,j}^{(1)}) \Delta t, \\
\tilde{U}_{i,j}^{(n+1)} &= \frac{1}{3} \tilde{U}_{i,j}^{(2)} + \frac{2}{3} \tilde{U}_{i,j}^{(1)} - \frac{2}{3} \mathcal{R}(\tilde{U}_{i,j}^{(2)}) \Delta t.
\end{align*}
\]

In particular, we computed the spatial residuals at stage \(s\),

\[
\mathcal{R}(\tilde{U}_{i,j}^{(1)}) = \frac{1}{\Delta x} (\tilde{F}_{i+1/2,j}^{(1)} - \tilde{F}_{i-1/2,j}^{(1)})
+ G_{i+1/2,j}^{(1)} - G_{i-1/2,j}^{(1)}
+ H_{i+3/2,j}^{(1)} - H_{i-3/2,j}^{(1)}
- S_{i,j}^{(1)},
\]

using the numerical techniques described in Sects. 2.3 and 2.4.

Finally, in order to achieve numerical stability, we limited the time step according to

\[
\Delta t = \frac{\text{CFL}}{N_{\text{dim}} \min_{i,j,k} \left( \frac{\Delta x}{|V_{i,j,k} + c_{i,j,k}|} \right)},
\]

where \(N_{\text{dim}}\) is the number of spatial dimensions. In all the tests presented in Sect. 3, we always adopted CFL = 0.8. We preferred to use a third-order accurate time stepper over less computationally expensive (but more inaccurate) methods, such as the "Midpoint rule" or SSP-RK2 (Shu & Osher 1988), so that the spatial instead of the temporal discretization would contribute most to the building up of global truncation errors.

3. Convergence properties of different Godunov-type methods

In this section, we check if the methods included in our study (described in Sect. 2) converge to the same numerical solution for several physical quantities of interest. In particular, we test a Kelvin-Helmholtz instability and a more complex setup characterized by the presence of turbulent convective flows, turbulent entrainment, and wave excitation. The results of the latter set of simulations allow us to estimate the computational cost per fixed accuracy for any given scheme, which we show in Sect. 4.

3.1. Kelvin–Helmholtz instability

We first test all of our 18 combinations of numerical schemes as described in Sects. 2.3 and 2.4 on a 2D Kelvin–Helmholtz problem with the initial condition

\[
\begin{align*}
\rho &= \gamma, \\
u &= M_0 \left[ 1 - 2\eta(y) \right], \\
v &= \frac{M_0}{10} \sin(2\pi x), \\
p &= 1, \\
X &= \eta(y),
\end{align*}
\]

where \(\gamma = 1.4\) and

\[
\eta(y) = \begin{cases} 
\frac{1}{2} \left[ 1 + \sin \left( 16\pi (y + 0.25) \right) \right], & \text{for } y > -\frac{5}{2} \text{ and } y < -\frac{7}{2}, \\
1, & \text{for } y \geq \frac{7}{2} \text{ and } y \leq \frac{9}{2}, \\
\frac{1}{2} \left[ 1 - \sin \left( 16\pi (y - 0.25) \right) \right], & \text{for } y > \frac{9}{2} \text{ and } y < \frac{7}{2}, \\
0, & \text{otherwise}.
\end{cases}
\]

The smooth function \(\eta(y)\) provides a resolvable transition between layers moving in opposite horizontal directions. The initial speed of sound is unity, so \(M_0\) is a tunable initial Mach number of the shear flow. We discuss solutions with \(M_0 = 10^{-1}, 10^{-2},\) and \(10^{-3}\). In Eq. (84), a smooth initial perturbation with an amplitude of \(M_0/10\) is included as a velocity component perpendicular to the shear flow. The computational domain, assumed to be periodic in both \(x\) and \(y\), spans \(0 \leq x, y \leq 2\), \(-0.5 \leq y \leq 0.5\).

The fact that the transition function \(\eta(y)\) between the shear layers is smooth\(^8\) allows us to compute numerically converged solutions even in the absence of physical viscosity as long as the simulations are stopped before the flow field becomes chaotic (see also Robertson et al. 2010; McNally et al. 2012; Lecoanet et al. 2017; Berlok & Pfrommer 2019). Each of the two transitions spans only \(1/16\) of the domain height and is poorly resolved on the coarser grids used in our tests. Therefore, we improve the accuracy of the initial cell averages that involve \(\eta(y)\) by averaging \(\eta(y)\) over 100 points uniformly distributed in the \(y\)-range covered by each cell. We measure numerical errors with respect to a reference solution computed using PSH reconstruction and the LHLLC flux function on a \(8192 \times 4096\) grid. The solution for \(M_0 = 10^{-2}\) is shown in Fig. 2 at four points in time\(^9\). As the instability grows in amplitude, the sinusoidal initial perturbation is rolled up into a series of vortices. Parts of the initial shear layers are stretched and become trapped in the centers of the vortices. Other parts of the shear layers become substantially narrower. We quantify this phenomenon by computing the minimum scale height \(\min(H_x) \equiv 1/\max(|V_x|)\) of the passive scalar \(X\). Figure 3 shows that this quantity drops by as much as a factor of 28 between \(t = 0\) and \(t = 0.8M_0^{-1}\). At the

\(^8\) \(\eta(y)\) and its first derivative are continuous but the second derivative is not.

\(^9\) We give the time in units of \(M_0^{-1}\) such that the same numerical value corresponds to the same evolutionary stage of the instability at all three initial Mach numbers we use.
Fig. 2. Reference solution to the Kelvin–Helmholtz problem with initial Mach number $M_0 = 10^{-2}$. The solution was computed using PSH reconstruction and the LHLLC flux function on an $8192 \times 4096$ grid. The mass fraction $X$ of the passive scalar is shown at four points in time: late in the linear growth of the instability ($t = 0.2 M_0^{-1}$), at an early stage of nonlinear evolution ($t = 0.4 M_0^{-1}$), at a stage when the primary vortices have fully formed ($t = 0.8 M_0^{-1}$); the final time for all of our other Kelvin–Helmholtz simulations, and at a late stage when fine threads have formed inside the primary vortices ($t = 1.6 M_0^{-1}$). We use the same color scale as in Figs. 4, D.2, and D.1, although $0 \leq X \leq 1$ in the reference solution.

Fig. 3. Resolution dependence of the minimum scale height $\min(H_x)$ of the passive scalar in the Kelvin–Helmholtz problem expressed in units of the computational cell width and shown at five points in time. Grid resolution is given by the number $N_x$ of computational cells along the x-axis. The initial Mach number is $M_0 = 10^{-2}$ and we use PSH reconstruction and the LHLLC flux function in this series of simulations. Once the steepest gradients in $X$ become resolved the minimum scale height starts to follow the linear scaling relations shown.

latter point in time, the minimum scale height is only 5.75 computational cells on the $8192 \times 4096$ reference grid. Extremely thin and difficult-to-resolve filaments appear at even later times (see Fig. 2). Therefore, we compare the solutions at $t = 0.8 M_0^{-1}$, making the problem nonlinear and challenging enough but not computationally prohibitive. The maximum Mach number in the flow field is $1.8 M_0$ at this point in time.

Figure 4 compares the distributions of the passive scalar in simulations with $M_0 = 10^{-2}$ computed on a $128 \times 64$ grid. The steepest gradients are strongly under-resolved on this grid (see Fig. 3), which increases the amplitude and visibility of small-scale artifacts produced by different methods. All of the six reconstruction functions lead to extremely diffusive solutions with the RUSANOV flux, although high-order methods with limiters (PPM84, PPM08, PSH) still preserve steep gradients at some places. Numerical diffusion is strongly suppressed with the LHLLC flux function owing to the explicit treatment of the contact wave in LHLLC. The three highest-order methods (PPM84, PPM08, PSH) reproduce the structure of the primary vortices (c.f. Fig. 2) much more closely than the lower-order methods (LIN, LIN+VL, PAR). However, the two PPM methods develop secondary instabilities around the primary vortices. This effect, not present in the reference solution, occurs also with the LHLLC flux function (see also McNally et al. 2012). The secondary instabilities become the dominant source of numerical errors. We find that these instabilities tend to grow when excessive velocity shear is generated at the grid scale and there is not enough numerical dissipation to suppress their growth. Therefore, the excitation of artificial, short wavelength Kelvin–Helmholtz instabilities is favored on coarser grids, which generate larger shear at the grid scale across the poorly resolved slip line, and by less dissipative Riemann solvers. For the same reason, simulations run with the LHLLC solver are more prone to developing secondary instabilities at higher rather than lower Mach numbers, as LHLLC introduces less numerical diffusion into the system when modeling faster flows (see, e.g., the panels for PPM84+HLLC in Fig. 4 and Fig. D.1). As the grid is refined, the shear layers are progressively better resolved thus reducing grid scale shear and suppressing the growth of the secondary instabilities. Not surprisingly, the smooth interiors of the primary vortices are best represented with the highest-order method PSH (c.f. Fig. 2). Figures D.1 and D.2, respectively, show that the differences between the three flux functions become smaller with $M_0 = 10^{-1}$ and much larger with $M_0 = 10^{-3}$. This is expected because the amount of numerical dissipation (relative to the flow of interest) introduced by the RUSANOV and LHLLC flux functions increases with decreasing Mach number of the flow, see Sect. 2.4. With $M_0 = 10^{-3}$, only the PSH method reproduces the basic structure of the primary vortices when combined with the LHLLC flux. Comparing Fig. 4 with Fig. D.2, we see that the LHLLC flux produces results independent of the initial Mach number $M_0$ because all of the flows are considerably subsonic. All of the methods we test converge to the same flow pattern with $M_0 = 10^{-2}$, see Fig. D.7 for solutions computed on the $4096 \times 2048$ grid.

The magnitude of the largest overshoot (or undershoot) in the passive scalar is shown in the insets in Figs. 4, D.1, and D.2. Thanks to their use of limiters, the LIN+VL, PPM84, PPM08, and PSH methods largely eliminate the overshoots, making the methods useful for the advection of mass fractions. We observe only some accumulation of round-off-level overshoots (up to $\approx 10^{-14}$) with LIN+VL, independently of the initial Mach number or grid resolution. The PPM84, PPM08, and PSH methods produce small but finite overshoots. At $t = 0.8 M_0^{-1}$, the magnitude of the largest overshoots is $\approx 10^{-4}$ with PPM84, $\approx 10^{-2}$ with PPM08, and $\approx 10^{-3}$ with PSH across all grids and all initial Mach numbers. However, they only occur in a few cells or groups of cells scattered across the computational grid, so the total mass affected by this effect is negligibly small. Moreover, the amplitude of such overshoots, drops to the round-off level in many of our simulations as the resolution is increased. Unlim-

10 Similar plots for all initial Mach numbers and all computational grids are available on Zenodo (https://zenodo.org/doi/10.5281/zenodo.16289090).
iterated methods produce substantially larger overshoots, reaching 8% and 17% for LIN and PAR, respectively. They are the result of the dispersion errors characteristic of linear schemes of 2nd or higher order, so they are nearly always present around sharp structures and affect a much larger amount of mass than the sporadic and isolated overshoots produced by PPM84, PPM08, and PSH. The benefits of the hybrid approach in PSH thus become obvious – secondary instabilities are suppressed (as compared with PPM84 and PPM08) thanks to the absence of limiters for dynamic variables while mass fractions, for which limiters are used, are well behaved.

We quantify numerical convergence of all of our methods by measuring $L_1$ errors with respect to the reference solution in the density $\rho$, kinetic energy $E_k$, associated with motions perpendicular to the initial shear flow, and mass fraction $X$ of the passive scalar. We define the relative $L_1$ error for any quantity $q$ as

$$L_1 = \frac{1}{\sigma_{ref}} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \left| q_{i,j} - q_{i,j}^{ref} \right|,$$

where $q^{ref}$ is the reference solution and the sums run over the whole computational grid of $N_x \times N_y$ cells. The error is normalised using the standard deviation $\sigma_{ref}$ of $q^{ref}$. This choice is motivated by the fact that density fluctuations are much smaller than the mean density in our setup. The reference solution is always re-binned from its original, 8192×4096 grid, by repeatedly averaging groups of 2×2 neighboring cells until the desired grid resolution is achieved.

The $L_1$ errors for the set of simulations with the initial Mach number $M_0 = 10^{-2}$ are shown in Fig. 5. We first focus on the kinetic energy $E_k$, and mass fraction $X$. The solutions computed using the LHLLC flux approach 2nd-order convergence, as expected. Only the PPM84 reconstruction method gives a slight decrease in the convergence rate in $E_k$, on the finest of our grids. The same effect is observed in the sets of simulations with $M_0 = 10^{-1}$ (Fig. D.3) and $M_0 = 10^{-3}$ (Fig. D.4). The convergence rate of PPM08 also decreases on very fine grids in the latter case. Apart from this, the convergence curves are nearly independent of the initial Mach number with the LHLLC flux (c.f. Figs. 5, D.3, and D.4), confirming its low-Mach property. The highest-order reconstruction methods usually (but not always) produce the smallest errors on a given and sufficiently fine grid. The errors differ by up to one order of magnitude, although the overall 2D scheme is 2nd-order accurate in all of the cases.

The magnitude of numerical errors strongly increases when we decrease initial Mach number with the non-low-Mach flux functions RUSANOV and HLLC. As described above, the spatial structure of errors produced by linear methods differs from that produced by methods with limiters. Figures 5, D.3, and D.4 show that the linear methods LIN, PAR, and for dynamic variables also PSH approach 2nd-order convergence on sufficiently fine grids even if the magnitude of the errors is much larger than what we obtain with the LHLLC flux function.

The two PPM methods converge poorly in $\rho$ at low Mach numbers (Figs. 5 and D.4). Because the initial density is constant everywhere, all of the density fluctuations are the integrated effects of the divergence (or convergence) of the velocity field in the continuity equation. The magnitude of the velocity divergence in simulations computed on the 512×256 grid is shown in Figs. 6, D.5, and D.6. The solutions computed using the LIN, PAR, and PSH methods clearly show the structure of the primary vortices with some oscillations in the shear layers and a background of relatively weak, large-scale sound waves. On the other hand, the solutions computed using the PPM84 and PPM08 methods show a large amount of small-scale numerical noise. After inspecting the time dependence of these artificial structures, we conclude that some of these structures travel with the flow while others have the character of small-scale sound waves.

We believe that both originate from cumulative effects of the limiters switching their local state many times over the time span of the simulation. The switching can result in rapid changes in the magnitude of the discontinuities at cell faces and, consequently, in the amount of dissipation applied in the Riemann solver (i.e., the numerical flux function). This effect is weakest with the low-dissipation flux function LHLLC but it is still clearly present on the 4096×2048 grid, see Fig. D.8. The structures disappear when we disable the limiters in PPM84 and PPM08 (not shown in the figures). We do not observe the fast-propagating, small-scale sound waves with LIN+VL but the method does produce thin structures in the velocity divergence around the primary vortices, which follow the flow.

Overall, the PSH method produces by far the smallest errors with the RUSANOV and LHLLC flux functions and initial Mach numbers $M_0 \leq 10^{-2}$ (Figs. 5 and D.4). The only exception is the poor convergence (or even divergence) in the passive tracer $X$ observed with the PSH+RUSANOV combination in simulations with $M_0 \leq 10^{-2}$ performed on very fine grids. This effect is also likely related to the limiters switching their local state many times over the time span of the simulation but we did not investigate it further.

### 3.2. 3D simulations of convection, turbulent entrainment, and wave excitation

In this section, we consider a test problem in which a variety of complex hydrodynamic phenomena can be captured on the same computational grid. These include convective transport of energy, turbulent mixing at a convective boundary, and the propagation of internal waves in a stably stratified layer, all of which are often encountered in simulations of geophysical and stellar hydrodynamics. The initial conditions for this test are adopted from the work of Andrassy et al. (2022) and they represent a hydrostatic solution of Eq. (2). In particular, the thermodynamic conditions in this setup are similar to those found in an oxygen-burning shell of a massive star. We map the hydrostatic stratification (see Fig. 7) on an evenly spaced, 3D, Cartesian grid defined by $(x, y, z) \in [-1, 1] \times [1, 3] \times [-1, 1]$. The gravitational acceleration points downward in the $y$-direction,

$$g_y = g_0 f_y(y)^{-5/4},$$

where $g_0 = -1.414870$ and

$$f_y(y) = \begin{cases} \frac{1}{2} \left[ 1 + \sin \left( 16\pi \left( y - \frac{1}{4} \right) \right) \right], & \text{for } 1 \leq y < 1 + \frac{1}{16}, \\
1, & \text{for } 1 + \frac{1}{16} \leq y \leq 3 - \frac{1}{16}, \\
\frac{1}{2} \left[ 1 + \sin \left( 16\pi \left( y - \frac{5}{4} \right) \right) \right], & \text{for } 3 - \frac{1}{16} < y \leq 3. \end{cases}$$

As in Andrassy et al. (2022), we work with rescaled quantities such that the speed of sound and the density at the base of the...
The initially hydrostatic stratification is described by a piecewise-polytropic relation in the form

\[
\frac{\partial \ln \rho}{\partial ln \rho} = \begin{cases} 
\gamma_0, & \text{for } 1 \leq y < 2 - \frac{1}{16}, \\
\gamma_0 + \eta(y) (y_1 - \gamma_0), & \text{for } 2 - \frac{1}{16} \leq y \leq 2 + \frac{1}{16}, \\
\gamma_1, & \text{for } 2 + \frac{1}{16} < y < 3,
\end{cases} 
\]  

(91)

where \(\gamma_0 = 5/3, \gamma_1 = 1.3,\) and \(\eta(y)\) is a smooth function,

\[
\eta(y) = \begin{cases} 
0, & \text{for } 1 \leq y < 2 - \frac{1}{16}, \\
\frac{1}{2} [1 + \sin(8\pi y)], & \text{for } 2 - \frac{1}{16} \leq y \leq 2 + \frac{1}{16}, \\
1, & \text{for } 2 + \frac{1}{16} < y < 3.
\end{cases} 
\]  

(92)

The adiabatic index is \(\gamma = \gamma_0\). In this work, we assume periodic boundary conditions in the horizontal \(x\)- and \(z\)-direction, whereas reflecting, stress-free boundaries are used in the vertical \(y\)-direction. This problem is set up such that turbulent convective flows develop in the lower half of the domain, which is initially hydrostatic, while internal gravity waves are free to propagate in the upper, stably stratified layer. To keep track of the position of the upper convective boundary, at \(t = 0\) we fill the stable layer with a passive scalar whose abundance smoothly turns to 0 across the upper boundary of the adiabatic region according to \(X = \eta(y)\).

In order to drive the convection, we include a time-independent heat source that continuously injects energy into the system close to the base of the box. The rate of energy released per unit volume,

\[
\dot{q}(y) = \begin{cases} 
\dot{q}_{\text{in}} \sin(8\pi y) \frac{\sin(4\pi \Delta x)}{(4\pi \Delta x)}, & \text{for } 1 \leq y < 1 + \frac{1}{8}, \\
0, & \text{for } 1 + \frac{1}{8} \leq y \leq 3,
\end{cases} 
\]  

(93)

is added to the right-hand side of Eq. (2) as

\[
\dot{S} \rightarrow \dot{S} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ q \end{bmatrix}. 
\]  

(94)

In the work of Andrassy et al. (2022), the amplitude of the heat source was \(\dot{q}_{\text{in}} = 3.79572 \times 10^{-4}\), which gave rise to convective flows with a root-mean-square Mach number \(M_{\text{rms}} = 0.04\). To make the problem more challenging, here we decrease \(\dot{q}\) by a factor of ten, so \(\dot{q}_{\text{in}} = 3.79572 \times 10^{-5}\). The lower heating rate, according to the well-established \(M \propto \dot{q}_{\text{in}}^{1/3}\) relation (see, e.g., Woodward et al. 2014; Käpylä 2021; Horst et al. 2021), should drive convection at \(M_{\text{rms}} \approx 0.02\).

In this setup, internal gravity waves (IGWs) are excited by the interaction of the convective flows with the bottom boundary of the stably stratified layer. The wavelength of IGWs in the direction of gravity becomes shorter when the waves are excited at progressively lower temporal frequencies (Sutherland 2010). At the heating rate we consider, the most prominent IGWs that originate at the convective boundary are only barely spatially resolved on the coarsest of our grids with 128³ cells. Therefore, we decide not to decrease \(\dot{q}_{\text{in}}\) even further for these simulations because it would give rise to convective flows with lower characteristic frequencies and lead to the generation of unresolved IGWs in the stable layer. Due to the fully compressible nature of SLH, we also expect short-wavelength sound waves to be generated, although at much lower amplitudes than those of IGWs at the typical Mach numbers encountered in this test problem (Lighthill 1952).

As done for the test described in Sect. 3.1, here we run simulations for each of the 18 considered combinations of Riemann solvers and spatial reconstruction schemes. To judge the numerical convergence of our results, each combination of methods is run on grids with 128³ and 256³ cells. Additionally, we run a single simulation on a 512³ grid using the LHLLC Riemann solver and the PAR reconstruction scheme, which we consider the reference solution for this test problem. However, because of the chaotic nature of the turbulent flows that arise in the convective layer, convergence is not expected in the exact flow morphology, so we do not compute \(L_1\) errors as done in Sect. 3.1. Instead, we analyze the convergence of the numerical results in terms of ensemble-averaged quantities that are representative of the dynamical properties of the system, such as kinetic energy spectra computed in the convective and stable layers.

To break the initial symmetry, we add a perturbation to the hydrostatic density stratification in the form

\[
\Delta \rho = 1.1 \times 10^{-5} \frac{9(y)}{q_{\text{in}}} [\sin(3\pi x) + \cos(\pi x)] [\sin(3\pi z) - \cos(\pi z)]. 
\]  

(95)

The subsequent evolution of the system and the development of convection is shown in Fig. 8. The density perturbation, along-side the action of the heat source, generate packets of fluid with higher entropy content than the adiabatic surroundings. The packets of hot and low-density material buoyantly rise in the adiabatic stratification until they reach \(y \approx 2\). At this height, the temperature stratification turns subadiabatic and the buoyant force acting on the rising plumes changes sign, forcing them to overturn. IGWs excited at the bottom boundary of the subadiabatic region propagate upward in the stratification (with characteristic Mach numbers in the range from 0.005 to 0.01) and are subsequently reflected at the top boundary of the domain. Shear instabilities break the large-scale buoyant structures that arise in the adiabatic layer and initiate the cascade of kinetic energy toward smaller scales. Turbulent convection fully develops after approximately one convective turnover time scale, \(\tau_{\text{conv}} = 133\) time units, which we define according to

\[
\tau_{\text{conv}} = \frac{2L_{\text{conv}}}{\langle |\mathbf{v}|^2 \rangle}. 
\]  

(96)

In Eq. (96), the root-mean-square convective speed is averaged over several convective turnover time scales and \(L_{\text{conv}} = 1\) is taken as representative of the vertical extent of the convection zone. In our reference simulation, the root-mean-square Mach number in the convection zone is \(M_{\text{rms}} = 0.019 \pm 0.001\), which is in agreement with the value predicted by the \(M_{\text{rms}} \propto \dot{q}_{\text{in}}^{1/3}\) scaling relation. All simulations are run until \(t_{\text{max}} = 32\tau_{\text{conv}}\) to have a proper coverage of the dynamical evolution of the system and to compute meaningful time averages needed for the following analysis.

3.2.1. Flow morphology and spatial kinetic energy spectra

In Fig. 9, we show snapshots of the Mach number taken at the vertical plane \(z = 0\), obtained at the final time \(t = 32\tau_{\text{conv}}\).

The error bar represents one standard deviation computed over the last 20 convective turnovers.
with all the methods tested here. Although all panels show results computed using the same grid resolution (256^3), there is a vast difference between the methods in terms of effects induced by numerical dissipation. With the RUSANOV flux, convection mainly happens in the form of large-scale plumes and IGWs are barely excited in the stable layer except when high-order methods such as PSH or PPM08 are used to reconstruct the Riemann states. Smaller-scale structures typical of turbulent flows become progressively more visible in the convection zone with higher-order reconstruction methods and less dissipative solvers. Internal waves with progressively shorter wavelengths also tend to be more visible in the stable layer with less dissipative methods.

The effects of the numerical dissipation on the properties of the flow are better quantified in Fig. 10 and 11, where we show the power spectra of kinetic energy extracted from a horizontal plane in the middle of the convection zone (\(y = 1.5\)) and stable layer (\(y = 2.5\)) as functions of the horizontal wavenumber, \(k_h = \sqrt{k_x^2 + k_z^2}\). We compute \(k_x\) and \(k_z\) as

\[
k_x = \begin{cases} m, & \text{for } 0 \leq m \leq \left\lfloor \frac{N_z-1}{2} \right\rfloor, \\ -N_x + m, & \text{for } \left\lfloor \frac{N_z-1}{2} \right\rfloor < m < N_x, \end{cases}
\]

\[
k_z = \begin{cases} n, & \text{for } 0 \leq n \leq \left\lfloor \frac{N_x-1}{2} \right\rfloor, \\ -N_z + n, & \text{for } \left\lfloor \frac{N_x-1}{2} \right\rfloor < n < N_z, \end{cases}
\]

where \(\lfloor \cdot \rfloor\) represents the floor function and \(N_x\) and \(N_z\) are the numbers of cells in the \(x\)- and \(z\)-direction, respectively. The spectra are averaged over the time interval \(t \in (10\tau_{\text{conv}}, 32\tau_{\text{conv}})\).

In the convection zone, all of the curves approximately agree with the Kolmogorov scaling law \(k^{-5/3}\) on some intermediate scales. We notice that the kinetic energy spectra shown in Fig. 10 present a pile-up of kinetic energy at the bottom of the inertial range. This phenomenon, also known as “bottleneck effect,” is often observed both in hydrodynamic simulations (see, e.g., Dobler et al. 2003) and experimental studies (see, e.g., Küchler et al. 2019) of turbulent flows. The extent of the inertial range of the turbulent kinetic energy spectrum greatly differs from method to method. With a fixed spatial reconstruction scheme, the kinetic energy spectrum obtained with LHHLC enters the viscous subrange (where the kinetic energy of the turbulent eddies is dissipated into internal energy of the gas) at higher wavenumbers as compared to both HLLC and RUSANOV. Reconstruction schemes based on slope limiters are characterized by shallower spectra in the viscous subrange than those generated by unlimited schemes. In the stable layer (see Fig. 11), RUSANOV and HLLC generate much weaker IGWs than the reference run even on large scales, except when used in combination with PSH.

Both in the convective and stable layers, all spectra converge to the reference solution if the order of the spatial reconstruction method or the grid resolution are increased.

### 3.2.2. Turbulent entrainment at the convective boundary

The overturning of the convective flows at the upper convective boundary gives rise to a variety of hydrodynamic processes (including shear instabilities, convective overshoot, and breaking of surface waves) that lead to the entrainment of material from the upper, stably stratified layer into the convection zone. The conglomeration of these processes is known in the literature as “convective boundary mixing” (see, e.g., Meakin & Arnett 2007; Andreasson et al. 2020; Horst et al. 2021; Anders & Pedersen 2023) and it increases the size of convective regions over time. As part of our comparison study, we quantify the impact of the choice of a specific combination of methods in Godunov-type schemes on the evolution of the convective boundary. At any given point in time, we assume that the vertical coordinate of the boundary between the convective and stable layer, \(y_{\text{cb}}\), is the position at which the horizontal average of the passive scalar \(X\) takes the value

\[
\bar{X} = \frac{1}{2}(X_{\text{min}} + X_{\text{max}}),
\]

where \(X_{\text{min}}\) and \(X_{\text{max}}\) are the minimum and maximum values of \(X\), respectively. Such a choice for \(y_{\text{cb}}\) is justified by the fact that the initial abundance of the passive scalar, \(X\), smoothly transitions from 0 to 1 at \(y \approx 2\), which is the boundary of the initially adiabatic layer where convection sets in first. The time evolution of \(y_{\text{cb}}\) is shown in Fig. 12 for all of our combinations of methods. We note that PPM-based methods used in combination with RUSANOV do not show signs of numerical convergence to the reference solution. Also, the other reconstruction schemes used with RUSANOV seem to accelerate the mixing at the convective boundary with respect to the reference run, especially on the 128^3 grid. In reality, this is an artifact of the method used to estimate the position of the convective boundary \(y_{\text{cb}}\). In fact, because RUSANOV does not resolve shear or contact waves (see Sect. 2.4), the initial gradient in \(X\) at the convective boundary is further smoothed out by the effects of numerical dissipation. When convection fully develops, it mixes the passive scalar that has diffused inside the convection zone and homogenizes its abundance, thus shifting the formal position of \(X = \bar{X}\) outward. This effect is further enhanced with the most diffusive reconstruction methods tested here. Therefore, with the RUSANOV solver, entrainment of material from the stable to the convective layer is mostly determined by numerical diffusion rather than turbulent mixing and the distribution of the passive tracer is not representative of \(y_{\text{cb}}\). On the other hand, all of the other methods tested here quickly achieve numerical convergence to the reference solution with increasing grid resolution, the order of the reconstruction scheme, or by using progressively less dissipative Riemann solvers. Among the data sets obtained with the six reconstruction schemes, the maximum relative deviation of \(y_{\text{cb}}\) from the reference solution at the final time is 20% and 15% with HLLC on the 128^3 and 256^3 grids, respectively, while with LHHLC it decreases from 12% to 5% with increasing grid resolution.

### 3.2.3. Shape of the convective boundary

The properties of the upper convective boundary are also studied by means of horizontal averages in the Brunt-Väisälä frequency, which determines both the spectrum of convectively driven IGWs (see, e.g., Sutherland 2010) and the rate of mass entrainment from the stable layer according to the bulk-Richardson-number mixing model (see, e.g., Meakin & Arnett 2007; Andreasson et al. 2020; Rizzuti et al. 2023). Here, the Brunt-Väisälä frequency is computed as

\[
N_{BV} = \left[ -\frac{g}{H_p} \left( \nabla \phi - \nabla \right) \right]^{1/2},
\]

where

\[
\nabla = \frac{d \ln T}{d \ln p},
\]

(101)
\[ T = p/\rho \] is the temperature of the gas\(^{14} \), \( \nabla_h = - (\text{d} \ln p/\text{d}y)^{-1} \) is the local pressure scale height, and \( \nabla_{\text{ad}} = 0.4 \) is the adiabatic temperature gradient. We average the profiles of \( N_{\text{BV}} \) over \( t \in (30 \tau_{\text{conv}}, 32 \tau_{\text{conv}}) \) so that the change in the position of the convective boundary is smaller than its width. At the same time, the chosen averaging time interval is long enough to partly suppress the statistical fluctuations induced by the turbulent nature of the convective flows. The results are shown in Fig. 13. As convection sets in and entrains material from the upper, stably stratified layer, it steepens the vertical entropy gradient \( \text{d} s/\text{d}y \) across the convective boundary. Therefore, due to the \( N_{\text{BV}}^{2} \propto \text{d}s/\text{d}y \) proportionality (see, e.g., Maeder 2009), the vertical profile of the Brunt-Väisälä frequency develops a spike in the region close to the convective boundary and it quickly approaches \( N_{\text{BV}}^{\text{grad}} \) in the rest of the stable layer. The fact that, in this setup, \( N_{\text{BV}} \) is characterized by large vertical gradients near the convective boundary (at least with HLLC and LHLLC), makes this quantity particularly suited to measure the amplitude of numerical undershoots or overshoots generated by the methods included in our study. In the regions right above the position of the spike in \( N_{\text{BV}} \), limited reconstruction methods (i.e., LIN + VL, PPM84, and PPM08) do not generate undershoots below the curve \( N_{\text{BV}}^{0} \) except for dynamical fluctuations smaller than 2% induced by the flows in the stable layer. On the other hand, reconstruction schemes that do not use limiters for dynamic variables (i.e., LIN, PAR, and PSH) produce undershoots whose amplitude is considerably larger than that of the dynamical fluctuations. The values of the largest undershoots relative to \( N_{\text{BV}}^{0} \) are shown in the insets of Fig. 13. Overall, the amplitude of the undershoots increases with the order of the unlimited reconstruction method, and it ranges from \( \approx 7 \% \) with LIN to almost 40% with PSH. The profiles of \( N_{\text{BV}} \) obtained on the 128\(^3 \) grid, not shown here (but available on Zenodo\(^{15} \)), reveal that the amplitude of the undershoots slightly increases from the 128\(^3 \) to the 256\(^3 \) grid for LIN and PAR, although the undershoot region becomes narrower on progressively finer grids. We note that, despite the large-amplitude undershoots generated by some of our unlimited reconstruction methods, such numerical errors do not have any significant impact on the growth of the convection zone, as shown in Fig. 12. Nonetheless, they could still affect the frequency spectrum of IGWs in the stable layer, although the spectral features from the stable layer (see Fig. 11) do show any such effect and converge to the reference solution. If numerical undershoots are largely to be avoided whilst generating a steep convective boundary, PPM84 + LHLLC or PPM08 + LHLLC could be the methods of choice.

### 3.3. Frequency power spectra

Finally, in Fig. 14, we show the frequency power spectrum of the vertical velocity component, \( v \), obtained in the middle of the stable layer for all of our 18 Godunov-type schemes. In particular, the frequency power spectra shown in Fig. 14 are Fourier projections of an array\(^{16} \) containing the value of \( v \) at \( (x,y,z) = (-0.2, 2.5, -0.2) \) at each time step. There is a clear distinction between the frequency range corresponding to the regime of IGWs (\( \omega < N_{\text{BV}} \)), where all spectra are almost flat, and the high-frequency range (\( \omega > N_{\text{BV}} \), where only sound waves are nonevanescent) in which the spectra are characterized by a steep decrease in power (although non monotonic) toward higher frequencies. The forest of lines visible at frequencies in the range from \( 1 \) to \( 50 \) correspond to the resonant pressure modes of the cavity as predicted by the linear wave theory in the Cowling approximation (Aerts 2021, see also Fig. 15). In the frequency regime of IGWs, the power spectrum seems to converge to the reference solution when increasing the grid resolution or when using less dissipative Riemann solvers. A more quantitative analysis of the IGW spectrum would require substantially longer simulations, which would benefit from more time averaging as well as reach higher frequency resolution.

On the other hand, the high-frequency domain is well resolved even in simulations run on the 128\(^3 \) grid, and differences between the spectra are clearly visible. Overall, there is a large spread in power (by almost 20 orders of magnitude at the Nyquist frequency) among the different methods tested in our study, which does not decrease with grid resolution. Therefore, numerical convergence is not achieved at high frequencies. We note that the spectra with the highest power density in the high-frequency range are always those obtained with limited reconstruction methods, in particular PPM84 and PPM08, while unlimited reconstruction methods tend to generate much quieter spectra and that are closer to the reference solution. In the simulations of the Kelvin–Helmholtz instability shown in Sect. 3.1, we find that a power excess in high-frequency, short-wavelength sound waves is generated by the complex limiting procedure performed in PPM84 and PPM08 to reconstruct the Riemann states at cell interfaces. In this setup, a similar phenomenon is likely responsible for generating an acoustic power excess close to the Nyquist frequency, where the spread between the spectra is maximum. The power excess, however, is still large down to frequencies similar to that of the fundamental oscillation mode of the cavity \( \omega_{\text{c}} = 1.1 \). Also, PPM-based methods used in combination with HLLC excite much stronger resonant lines than those generated with LHLLC, and their peak power density is almost as high as the flat part of the spectrum associated with IGWs. When used with PPM84, both RUSANOV and LHLLC produce a broad feature in the frequency spectrum whose peak is at \( \omega_{\text{peak}} \approx 50 \) and 100 on the 128\(^3 \) and 256\(^3 \) grids, respectively. If these were sound waves, their wavelength \( \lambda \) would be

\[
\lambda = \frac{2\pi}{\omega_{\text{peak}}} c_{\text{sound}}(y = 2.5) \approx 2.7 \Delta x,
\]

which is close to the Nyquist frequency in space. Therefore, such a feature in the spectrum may be caused by odd-even cell decoupling traveling at the local speed of sound. Additional peaks are observed at even higher frequencies than the frequency of the least resolvable sound wave on the grid (with a wavelength of two cells), so these are most likely numerical artifacts. Tests performed with lower CFL factors of 0.4 and 0.2 show that the amplitude of these peaks is slightly reduced when using shorter time steps in the simulation, but the level of the continuum in the power spectrum remains essentially unaltered. In light of these results, we advise against using PPM-based methods in simulations of sound generation by low-Mach-number turbulence.

### 4. Performance metrics

After proving that all of the methods tested in our study converge to the correct solution for most of the physical quantities of interest,
est\textsuperscript{17}, we can now search for the most efficient way to generate a numerical solution at the desired accuracy. In principle, one could use the L\textsubscript{2} errors computed from the simulations of the Kelvin–Helmholtz instability (see Sect. 3.1) to find the resolution of the grid on which the scheme achieves a given level of accuracy. Then, the computational cost of the simulation run on such a grid can be estimated if the wall-clock time spent by the program to perform a single cell update is known. Here, we prefer to use the results from the test problem involving turbulent convective flows and wave excitation (see Sect. 3.2), which is more challenging than the Kelvin–Helmholtz instability test and much closer to a real application of stellar hydrodynamics. However, due to the chaotic nature of the turbulent flows that develop in the convection zone, convergence in the L\textsubscript{2} error norm cannot be achieved. Therefore, rather than measuring the computational cost per fixed accuracy in the sense of L\textsubscript{2} errors, we opt to measure the computational cost of a simulation run with a specific combination of numerical methods that achieves a given level of effective resolution of the turbulent flows, \( N_{\text{eff}} \). We estimate the effective resolution obtained on a given grid with \( N^3 \) cells as

\[ N_{\text{eff}}(N_c) = \frac{2L_{\text{conv}}}{\lambda_{\text{vis}}(N_c)}, \]

where \( \lambda_{\text{vis}}(N_c) \) is a characteristic dissipation length scale in the turbulent kinetic energy spectrum. In high-resolution schemes, the amount of kinematic viscosity introduced into the system is not fixed, but rather its value depends on the local dynamical properties of the flow and it is often found to be a steep function of the spatial wavenumber (see, e.g., Porter & Woodward 1994).

Therefore, \( \lambda_{\text{vis}}(N_c) \) cannot be defined uniquely. Here, we choose

\[ \lambda_{\text{vis}}(N_c) \approx \frac{2L_{\text{conv}}}{k_{10}(N_c)}, \]

as a representative value for the dissipation length scale, with \( k_{10}(N_c) \) being the spatial wavenumber at which the kinetic power spectrum rescaled by the Kolmogorov law drops by one dex from its maximum. In this approximation, the effective resolution is simply given by

\[ N_{\text{eff}}(N_c) = k_{10}(N_c). \]

To perform a more precise measurement of \( N_{\text{eff}}(N_c) \), both the array of wavenumbers and of the kinetic energy spectrum are linearly interpolated on a finer grid. The values of the effective resolution (rescaled by the grid resolution \( N_c \)) obtained in the simulations included in our study are collected in Table 2. There is a clear trend toward higher effective resolution when using progressively higher-order reconstruction methods or less dissipative Riemann solvers.

By combining the values of \( N_{\text{eff}}(N_c) \) and the average wall-clock time \( \delta(t)(N_c) \) spent by the program to perform a single cell update, shown in Table 3, it is possible to estimate the computational effort required by each combination of numerical options to achieve the effective resolution of PPM08 + LHLLC\textsuperscript{18} as

\[ \Theta(N_c) = \frac{\delta(t)(N_c)_{\text{PPM08 + LHLLC}}}{\delta(t)(N_c)_{\text{PPM08 + LHLLC}}} \left( \frac{N_{\text{eff}}(N_c)}{N_{\text{eff, PPM08 + LHLLC}(N_c)}} \right)^{-4}. \]  

\textsuperscript{17} A prominent example of a nonconvergent numerical solution is the frequency power spectrum of sound waves in the test problem of turbulent convection and mass entrainment shown in Sect. 3.2.

\textsuperscript{18} In the following expression, we make use of the fact that the complexity of a 3D, Godunov algorithm, such as that implemented in SLH, scales with \( N^3 \), with \( N \) being the number of grid cells per dimension.

Equation (106) is rescaled such that the cost of the simulation run with PPM08 + LHLLC on any given grid is unity. All values of \( \Theta(N_c = 128) \) and \( \Theta(N_c = 256) \) are shown in Table 4. The spread of almost four orders of magnitude in the relative computational cost among the different methods is due to the steep dependence of \( \Theta(N_c) \) on \( N_{\text{eff}}(N_c) \), which varies by as much as a factor of 10 (see Table 2). On the other hand, \( \Theta(N_c) \) only scales linearly with \( \delta(t)(N_c) \), which in our simulations varies at most by a factor of \( \approx 3 \) (with LIN + RUSANOV and PPM84 + LHLLC achieving the lowest and highest wall-clock time per cell-update, respectively). The most expensive combination of methods among those tested in our study (in terms of computational cost per fixed resolving power) is LIN + VL + RUSANOV, \( \sim 1000 \) times as expensive as PPM08 + LHLLC. The strong numerical dissipation generated by RUSANOV leads to very poor performance of the finite-volume scheme even when used in combination with PPM-based reconstruction methods. Only with the unlimited PSH method such a flux function is capable of achieving acceptable performance (\( \Theta(N_c = 4) \)). When the reconstruction scheme is kept the same, using progressively less dissipative Riemann solvers decreases the cost of reaching the same effective resolution. At the typical Mach numbers encountered in the convection zone (\( M = 0.02 \)), the combination of LHLLC and second-order reconstruction schemes or PAR is considerably more expensive than PPM08 + LHLLC (\( \Theta(N_c) \) in the range from 10 to 30). The performance of the scheme increases when LHLLC is used with PPM-based reconstruction schemes (\( \Theta(N_c) \) in the range from 2 to 6) and it is even higher than that of PPM08 + LHLLC when coupled to PSH. PSH + LHLLC is the most performant method on both grids according to the chosen metric. Overall, the computational cost of the finite-volume scheme is considerably reduced when using the low-dissipation solver (by a factor from 2 to 10 with respect to a scheme using LHLLC and the same spatial reconstruction method). The relative performance of the scheme only varies slightly when LHLLC is used in combination with reconstruction methods less accurate than PSH, with the worst performance being achieved by LIN + VL (\( \Theta(N_c) \approx 3 \)).

We note that, for most methods, the value of \( \Theta(N_c) \) sensibly increases (by as much as a factor of \( \approx 2 \)) from the 128\textsuperscript{3} to the 256\textsuperscript{3} grid. Such differences are due to the small but systematic decrease of the rescaled effective resolution \( N_{\text{eff}}(N_c)/N_c \), with increasing the grid resolution (this behavior is also confirmed by the reference solution computed on the 512\textsuperscript{3} grid, see Table 2).

Although the differences observed in \( N_{\text{eff}}(N_c)/N_c \) among the two grids are at most 20%, they are significantly amplified after applying the steep scaling relation between \( \Theta(N_c) \) and \( N_{\text{eff}}(N_c) \). However, this effect is negligible if compared to the large spread obtained in the values of \( \Theta(N_c) \) on a given grid. Furthermore, we stress that our measure of \( \Theta(N_c) \) is based on a crude approximation of effective resolution of the turbulent flows, so the values provided in Table 4 should only be taken as estimates of the relative performance of different Godunov-type methods in simulations of turbulent convection. The measurements of the absolute performance provided in Table 3, which are needed to estimate \( \Theta(N_c) \), may also depend on the parallelization strategy and the number of cores used to run the simulations, especially for the methods that require many ghost cells (e.g., PPM08 and PSH) and are therefore characterized by higher communication costs.

5. Summary and conclusions

High-resolution, finite-volume schemes are popular methods for simulating the behaviour of astrophysical fluids. There is a wide range of spatial-reconstruction schemes, numerical flux...
Table 2. Effective resolution $N_{eq}(N_x)$ as defined in in Sect. 3.2 for all of the 18 Godunov-type methods tested in our study on grids with $N_x = 128$ and $N_x = 256$ cells per dimension. Here, all numbers are given as a percentage of the grid resolution $N_x$ for an easier comparison between different grids. In this metric, the rescaled effective resolution of the reference run ($\text{PAR} + \text{LHLLC} + 512^3$) is 24.43.

<table>
<thead>
<tr>
<th>Grid Size</th>
<th>Method</th>
<th>RUSANOV</th>
<th>HLLC</th>
<th>LHLLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>128$^3$</td>
<td>LIN</td>
<td>6.21</td>
<td>14.45</td>
<td>24.68</td>
</tr>
<tr>
<td></td>
<td>LIN+VL</td>
<td>5.24</td>
<td>13.53</td>
<td>21.97</td>
</tr>
<tr>
<td></td>
<td>PAR</td>
<td>7.04</td>
<td>16.68</td>
<td>29.05</td>
</tr>
<tr>
<td></td>
<td>PPM84</td>
<td>10.62</td>
<td>24.15</td>
<td>31.47</td>
</tr>
<tr>
<td></td>
<td>PPM08</td>
<td>14.21</td>
<td>28.42</td>
<td>33.85</td>
</tr>
<tr>
<td></td>
<td>PSH</td>
<td>20.47</td>
<td>37.24</td>
<td>49.22</td>
</tr>
<tr>
<td>256$^3$</td>
<td>LIN</td>
<td>5.47</td>
<td>12.61</td>
<td>23.30</td>
</tr>
<tr>
<td></td>
<td>LIN+VL</td>
<td>4.38</td>
<td>11.08</td>
<td>19.75</td>
</tr>
<tr>
<td></td>
<td>PAR</td>
<td>6.35</td>
<td>14.28</td>
<td>26.16</td>
</tr>
<tr>
<td></td>
<td>PPM84</td>
<td>10.10</td>
<td>21.23</td>
<td>29.06</td>
</tr>
<tr>
<td></td>
<td>PPM08</td>
<td>13.55</td>
<td>27.00</td>
<td>32.02</td>
</tr>
<tr>
<td></td>
<td>PSH</td>
<td>19.90</td>
<td>35.08</td>
<td>47.00</td>
</tr>
</tbody>
</table>

Table 3. Mean wall-clock time spent by the program to advance the solution by one time step with RK3 in a single cell of the computational grid, in units of $\mu$s. The numbers provided in the tables are averages of five measurements, each of which is obtained by evolving the setup described in Sect. 3.2 for 100 steps. Relative errors in the averages (in the sense of 1\%e) are smaller than 2\% in all cases. Every simulation run for this analysis is MPI parallelized using grids with $32^3$ cells per task. Computations are performed on 2.3 GHz Intel Xeon, Skylake-based processors.

<table>
<thead>
<tr>
<th>Grid Size</th>
<th>Method</th>
<th>RUSANOV</th>
<th>HLLC</th>
<th>LHLLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>128$^3$</td>
<td>LIN</td>
<td>1.47</td>
<td>1.68</td>
<td>1.70</td>
</tr>
<tr>
<td></td>
<td>LIN+VL</td>
<td>1.57</td>
<td>1.80</td>
<td>1.89</td>
</tr>
<tr>
<td></td>
<td>PAR</td>
<td>1.47</td>
<td>1.74</td>
<td>1.78</td>
</tr>
<tr>
<td></td>
<td>PPM84</td>
<td>4.12</td>
<td>4.30</td>
<td>4.31</td>
</tr>
<tr>
<td></td>
<td>PPM08</td>
<td>3.57</td>
<td>3.75</td>
<td>3.83</td>
</tr>
<tr>
<td></td>
<td>PSH</td>
<td>2.01</td>
<td>2.36</td>
<td>2.41</td>
</tr>
<tr>
<td>256$^3$</td>
<td>LIN</td>
<td>1.60</td>
<td>1.73</td>
<td>1.77</td>
</tr>
<tr>
<td></td>
<td>LIN+VL</td>
<td>1.73</td>
<td>1.87</td>
<td>1.88</td>
</tr>
<tr>
<td></td>
<td>PAR</td>
<td>1.60</td>
<td>1.81</td>
<td>1.86</td>
</tr>
<tr>
<td></td>
<td>PPM84</td>
<td>4.23</td>
<td>4.40</td>
<td>4.47</td>
</tr>
<tr>
<td></td>
<td>PPM08</td>
<td>3.79</td>
<td>4.03</td>
<td>4.00</td>
</tr>
<tr>
<td></td>
<td>PSH</td>
<td>2.24</td>
<td>2.47</td>
<td>2.52</td>
</tr>
</tbody>
</table>

functions and time-integration methods that can be combined into a numerically stable and robust, finite-volume scheme. Focusing on subsonic flows, we have analyzed the accuracy and computational cost of all possible combinations of six spatial-reconstruction schemes and three numerical flux functions, meaning 18 methods in total. The numerical solution was marched in time with a semi-discrete scheme based on a third-order Runge–Kutta method. This choice was motivated by our focus on the spatial accuracy of the schemes and the related need to suppress time-stepping errors.

We considered two main test problems. The first is a Kelvin–Helmholtz instability problem, in which the initial shear flows have Mach numbers of $10^{-1}$, $10^{-2}$, and $10^{-3}$. We used a smooth initial condition to make it possible to obtain numerically convergent solutions to the inviscid Euler equations at a fixed point in time in the nonlinear phase of the instability. We characterized the accuracy of the solutions by (i) performing a qualitative assessment of the spatial structure of the solutions and (ii) by measuring $L_1$ errors with respect to a reference solution obtained on a fine grid of $8192 \times 4096$ cells.

The low-Mach function LHLLC is found to be much less dissipative and much more accurate than the HLLC and RUSANOV flux functions at the two lowest Mach numbers considered. Even though the overall 2D numerical scheme is 2nd-order accurate, the errors produced by different spatial reconstruction schemes span as much as an order of magnitude. Unlimited reconstruction schemes of increasing order, up to 7th, provide progressively more accurate results when the solution is sufficiently well resolved. However, these schemes are of limited use for the advection of mass fractions due to the oscillations and overshoots they produce around discontinuities. This issue is largely eliminated by the use of limiters in the schemes LIN+VL, PPM4, and PPM8. However, we find that the switching behaviour of the limiters introduces spurious structures and small-scale sound waves to the solutions, which severely reduce the accuracy of the methods when applied to slow flows. Our hybrid method PSH, which combines unlimited, 7th-order reconstruction for dynamic variables with PPM8 for mass fractions, is found to provide the most accurate solutions in nearly all of our simulations of the Kelvin–Helmholtz instability.

Our second test problem, chosen to be as close as possible to practical applications in the dynamics of stellar interiors, involves stratified turbulent convection generating waves in and entraining mass from an overlying stably stratified layer. The typical Mach number of the convection is $\approx 0.02$. The turbulent nature of the convective flow makes the solutions chaotic but space- and time-averaged quantities can still be meaningfully compared between different simulations. We ran the simulations on grids of $128^3$ and $256^3$ cells with one additional simulation on a $512^3$ grid serving as a reference solution.

Qualitatively speaking, the solutions match the trends seen in the simulations of the Kelvin–Helmholtz instability, that is to say combinations of the least dissipative flux functions with...
the highest-order reconstruction schemes provide the highest resolving power in both the convective and stable layers. Spatial spectra of kinetic energy computed in the convective and stable layers converge toward the reference solution with any of the 18 methods but the least dissipative combinations (especially PSH+LHLLC) are much closer to the reference solution than the most dissipative ones (especially LIN+VL+Rusanov). The spatial spectrum in both layers obtained with PSH+LHLLC on a $256^3$ grid is essentially the same as that with PAR+LHLLC on a $512^3$ grid.

The growth rate of the convective layer converges upon grid refinement with all 6 reconstruction schemes and both the HLLC and LHLLC flux function but the convergence is significantly faster with the low-Mach flux function LHLLC. The Rusanov flux function produces extremely viscous flows and large deviations from the reference solution in terms of the position of the upper convective boundary.

We show that schemes that do not use limiting for dynamic variables (LIN, PAR, PSH), unlike those that do (LIN+VL, PPM84, PPM08), produce overshoots in the Brunt-Väisälä frequency at the relatively sharp convective boundary. The magnitude of the overshoots increases with increasing order of accuracy of the method, as expected. Neither the growth rate of the convective layer, nor the temporal and spatial spectra of internal gravity waves are affected by this phenomenon in our simulations but we recommend careful testing when such high-order methods are applied to other setups or used to derive quantities not investigated here.

Temporal spectra of the vertical component of velocity in the stable layer show that all six reconstruction schemes produce qualitatively similar, flat and featureless spectra in the regime of internal gravity waves. On the other hand, the spectra of pressure (i.e., high frequency) waves are sensitive to the choice of the reconstruction scheme even with the low-Mach flux function LHLLC. The spectra also reinforce our conclusions based on the Kelvin–Helmholtz problem that methods that apply limiters to dynamic variables generate much more acoustic noise than methods that do not. This includes the popular methods PPM84 and PPM08. Although we do not know what the acoustic spectrum in our problem should look like, we show that the frequencies of vertical resonant modes agree with 1D linear wave theory.

Finally, we used the steep drop-off of the spatial kinetic energy spectra in the convective layer close to the grid scale to determine a measure of effective resolution. Rescaling the wall-clock time of our $128^3$ and $256^3$ simulations, we estimated the computational cost of matching the same effective resolution using our 18 methods. The cost is found to span almost four orders of magnitude. The largest cost reduction comes from choosing the least dissipative flux function, which is LHLLC in our study. We show that the reconstruction schemes PSH, PAR, and PPM08, in order of increasing cost, are the cheapest at the same effective resolution.

Our study demonstrates that it is generally advisable to use low-dissipation Riemann solvers in Godunov-type finite-volume schemes for simulating low-Mach-number flows. The low-Mach fix $\phi$ in Eq. (76) can easily be implemented in any HLLC-type scheme without affecting the absolute performance of the code, and it reduces the amount of computation required to achieve the same accuracy as HLLC by a factor ranging from 2 to 10 at typical Mach numbers of $\approx 0.01$. At Mach numbers of $\approx 10^{-4}$ or $\approx 10^{-3}$, similar to those encountered in the convective cores of main sequence stars, the usage of a low-Mach solver decreases the computational cost per fixed accuracy with respect to a conventional Riemann solver by even larger factors (Leidi et al. 2022). Although low-dissipation Riemann solvers such as LHLLC bring clear advantages in simulations of subsonic flows when used in Eulerian codes, their robustness and accuracy properties in quasi-Lagrangian, moving-mesh schemes still has to be verified. We note that such schemes hold promise for a particularly significant improvement because mesh motions aim at reducing advection errors by minimizing the velocities relative to the cell interfaces. This naturally leads to low-Mach-number flows in the comoving frame, which are better modeled by low-Mach-number Riemann solvers. We are currently testing an implementation of the LHLLC/D solvers of Minoshima & Miyoshi (2021) in the moving-mesh MHD code AREPO (Springel 2010), which we will report in a forthcoming study.

On the other hand, the choice of the spatial reconstruction scheme depends on the physical application of interest. Overall, second-order reconstruction methods lead to the generation of considerably more inaccurate results than higher-order schemes when the Mach number of the flow is low. If numerical overshoots have to be suppressed whilst retaining high resolving power, PPM84 and PPM08 should be the methods of choice. However, higher-order unlimited reconstruction methods should be preferred over PPM-based methods in simulations of sound wave generation by subsonic turbulence because they generate much less numerical noise in the acoustic spectrum. Less oscillatory alternatives to the hybrid PSH method, such as the 3rd-order piecewise parabolic hybrid (PPH) or the 5th-order piecewise quartic hybrid (PQH) methods described in Appendix A, could offer the best compromise between the complete removal of numerical overshots and reduction of acoustic noise.

Finally, we note that there are complex astrophysical environments where both high- and low-Mach-number flows can coexist. This scenario often occurs, for instance, in accretion disks (see, e.g., Klessen & Hennebelle 2010), star forming regions (see, e.g., Wang et al. 2024), or outer layers of stars (see, e.g., Wedemeyer et al. 2017). To model such diverse flow regimes, the simple use of unlimited higher-order reconstructions and low-dissipation solvers may give rise to numerical instabilities, especially in the proximity of shock fronts. To cure this problem, an alternative approach based on the local dynamical properties of the flow is needed (see, e.g., Mignone et al. 2011; Fleischmann et al. 2020). One possibility is to add a selection criterion to the function that computes the flux at a cell interface such that the LHLLC solver is used if the Mach number of the flow is lower than a conservative threshold (e.g., $M \approx 0.6$), whereas a proper shock-capturing method (e.g., HLL or LHLLC) is chosen for modeling faster flows. A similar argument can be made for the choice of the spatial reconstruction scheme. To avoid introducing large oscillations in the state quantities near large, nonlinear discontinuities, schemes that are more dissipative and possibly closer to being TVD (such as LIN+VL, PPM84, PPM08, or WENO) should be preferred over high order unlimited methods.

Acknowledgements. The work of GL and FKR is supported by the German Research Foundation (DFG) through the grant RO 3676/3-1. We acknowledge support by the Klaus Tschira Foundation. This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy EXC 2018/1 - 390009498 (the Heidelberg STRUCTURES Excellence Cluster). The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer SuperMUC-NG at Leibniz Supercomputing Centre (www.lrz.de). PVF was supported by the U.S. Department of Energy through the Los Alamos National Laboratory (LANL). LANL is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the U.S. Department of Energy (Contract No. 80232138CNA000001). This work has been assigned a document release number LA-UR-23-33257.
Fig. 4. Distributions of the mass fraction $X$ of the passive scalar in simulations of the Kelvin–Helmholtz instability with the initial Mach number $M_0 = 10^{-2}$ on a $128 \times 64$ grid. The results are shown at $t = 0.8 M_0^{-1}$. Rows and columns show different reconstruction schemes and numerical flux functions, respectively. The color scale intentionally shows values $X < 0$ and $X > 1$ to highlight the overshoots that some of the schemes produce. The absolute value of the largest overshoot outside of this range is given in each panel.
Fig. 5. Relative $L_1$ errors for different variables (rows), flux functions (columns), and reconstruction methods (legend) as functions of the grid resolution in the simulations of the Kelvin–Helmholtz instability with the initial Mach number $M_0 = 10^{-2}$. The dashed and dotted lines, which are at the same locations in all of the panels, show the 1st- and 2nd-order scalings to guide the eye.
Fig. 6. Magnitude of velocity divergence in simulations of the Kelvin–Helmholtz instability with the initial Mach number $M_0 = 10^{-2}$ on a $512 \times 256$ grid. The results are shown at $t = 0.8 M_0^{-1}$. Rows and columns show different reconstruction schemes and numerical flux functions, respectively. The small-scale structure produced by the LIN+VL, PPM84, and PPM08 schemes, which does not occur with the LIN, PAR, and PSH schemes, is caused by the use of limiters.
Fig. 7. Hydrostatic stratification of density, gas pressure, pseudo-entropy \( \frac{p}{\rho^n} \), and gravity at \( t = 0 \) in the problem of turbulent convection and wave excitation at a convective boundary, based on the initial conditions provided in Andrassy et al. (2022). The distribution of the mass fractional abundance of a passive tracer \( X \) and the heat source \( \dot{q} \) are also shown, here represented by a dash-dotted pink and dashed black line, respectively. In this setup, the gravitational acceleration smoothly turns to zero at the bottom and top boundaries of the spatial domain to make the problem consistent with the usage of reflecting, stress free boundary conditions, which for the gas pressure imply \( \frac{\partial p}{\partial y} = 0 \).
Fig. 8. Pseudo-entropy ($A = p/\rho^n$) fluctuations with respect to the hydrostatic, background state computed at different times during the development of the convection, as indicated by the insets, in the problem of turbulent convection and wave excitation. Data are visualized in the $z = 0$ plane. This simulation was run using the PAR method and LHLLC on a $512^3$ grid and we consider it our reference solution. Convective plumes start exciting IGWs at the upper convective boundary by $t \approx \tau_{\text{conv}}$ (see center bottom panel).
Fig. 9. Snapshots obtained at the final simulation time ($t = 32\tau_{\text{conv}}$) showing the distribution of the Mach number at $z = 0$ in the simulations of turbulent convection and wave propagation in a 3D box. The grid resolution is $256^3$. Each panel is cut between $y = 1.3$ and $y = 2.7$ for a better visualization of the flows in the proximity of the upper convective boundary.
Fig. 10. Kinetic energy spectra computed on the horizontal plane $y = 1.5$ of the convective layer for all of our 18 combinations of Riemann solvers and spatial reconstruction schemes in the problem of turbulent convection and wave excitation. The spectra have been averaged over the time interval $t \in (10 \tau_{\text{conv}}, 32 \tau_{\text{conv}})$ and rescaled by the Kolmogorov law ($k^{-5/3}$) and by the value $\hat{E}_0$ of the spectral energy density of the reference run at $k_h = 15$. 

\begin{tabular}{ll}
LIN & PPM84 \\
\begin{tabular}{c}
\includegraphics[width=0.4\textwidth]{lin.png} \\
\includegraphics[width=0.4\textwidth]{lin_vl.png} \\
\includegraphics[width=0.4\textwidth]{par.png}
\end{tabular} & \begin{tabular}{c}
\includegraphics[width=0.4\textwidth]{ppm84.png} \\
\includegraphics[width=0.4\textwidth]{ppm08.png} \\
\includegraphics[width=0.4\textwidth]{psh.png}
\end{tabular}
\end{tabular}
Fig. 11. Kinetic energy spectra computed on the horizontal plane $y = 2.5$ of the stable layer for all of our 18 combinations of Riemann solvers and spatial reconstruction schemes in the problem of turbulent convection and wave excitation. The spectra have been averaged over the time interval $t \in (10 \tau_{\text{conv}}, 32 \tau_{\text{conv}})$. The black solid line represents the reference run (REF), which is computed using the LHLLC Riemann solver and the PAR reconstruction scheme on a $512^3$ grid.
Fig. 12. Time evolution of the position of the upper convective boundary \( (y_{cb}) \) in the problem of turbulent convection and entrainment described in Sect. 3.2 The top and bottom rows show results from 128\(^3\) and 256\(^3\) simulations, respectively. The black solid line represents the reference run (REF).
Fig. 13. Vertical profiles of the Brunt-Väisälä frequency ($N_{BV}$) averaged over the time interval $t \in (30\tau_{conv}, 32\tau_{conv})$ in the problem of turbulent convection and wave excitation. Each panel shows the results of simulations run using the same reconstruction scheme but different Riemann solvers. Here, we only show results from the $256^3$ grid to avoid cluttering the figures. The reference run (REF) is represented by a black dashed line. The black dash-dotted line is the profile of the Brunt-Väisälä frequency at $t = 0$, and the vertical black dotted lines represent the position of the convective boundary at the beginning and the end of the chosen averaging time interval in the reference run, $y = 2.168$ and $y = 2.174$, respectively. The percentages shown in the insets for simulations run with the HLLC and LHLLC solvers represent the amplitude of the largest undershoot in $N_{BV}$ relative to $N_{BV,t=0}$, in the spatial range $y \in (2.15, 2.40)$. The amplitude of the largest undershoot in the reference run is 12%.
Fig. 14. Frequency power spectrum of the vertical velocity component $v$ extracted in the middle of the stable layer at $(x, y, z) = (-0.2, 2.5, -0.2)$ over the time series $t \in (10\tau_{\text{conv}}, 32\tau_{\text{conv}})$ in the problem of turbulent convection and wave excitation. The left and right panels show results obtained on the $128^3$ and the $256^3$ grid, respectively. Each row of panels shows the results of simulations run using 6 reconstruction schemes with the same Riemann solver. The reference solution (indicated with REF) is the black curve. The convective turnover frequency ($\omega_{\text{conv}} = 2\pi/\tau_{\text{conv}}$), the Brunt-Väisälä frequency at $(x, y, z) = (-0.2, 2.5, -0.2)$ ($N_{\text{BV}}$), and the frequency of the fundamental oscillation mode of the cavity ($\omega_0 = 1.1$) are represented by the black dashed-dotted, dashed, and dotted lines, respectively.
**Fig. 15.** Frequency power spectrum of $v$ as in Fig. 14 but showing only the results of PPM08 + HLLC on the 256$^3$ grid. The vertical black dotted lines represent the resonant modes of the cavity which we derived from the theory of linear oscillations in Cowling approximation (see, e.g., Aerts 2021). The pink line at $\omega = 1.1$ is the frequency of the fundamental oscillation mode of the cavity.
Appendix A: The PPH and PQH methods

The idea of applying limiters to passive scalars (or mass fractions) only, which lead us to formulating the PSH method (Sect. 3.1), can be applied at any order of accuracy. In this section, we provide two methods that can be seen as lower-order alternatives to PSH. Their potential advantages include a lower amplitude of overshoots in dynamic quantities and the need for fewer ghost cells.

We start with the unlimited parabolic method PAR (Sect. 3.2) and apply a two-step limiter to passive scalars. The first step is defined by the variable replacements

\[ a_{i-1/2,L,R} \mapsto \text{median}(a_{i-1} - a_{i-1/2,L,R}, a_i). \]  

(A.1)

These replacements are followed by the application of the PPM84 limiter defined by Eq. (31). We call the resulting method piecewise parabolic hybrid (PPH). The PPH method is exact wherever \( a(\xi) \) is locally parabolic, 3\textsuperscript{rd}-order accurate for general but smooth functions \( a(\xi) \), and it requires two ghost cells at domain boundaries.

A more accurate method, which we call piecewise quartic hybrid (PQH), can be obtained by assuming that within cell \( i \) \( a(\xi) \) can be described by the quartic polynomial

\[ a(\xi) = \sum_{n=0}^{4} c_n (\xi - \xi_i)^n. \]  

(A.2)

The five coefficients \( c_n \) are uniquely determined by the requirement that the averages of \( a(\xi) \) in cells \( i - 2 \) to \( i + 2 \) equal \( \overline{a_{i-2:n}} \) for \( n = 0, 1, \ldots, 4 \). The reconstructed states are then obtained by evaluating Eq. (A.2) at \( \xi = \xi_i - 1/2 \) and \( \xi = \xi_i + 1/2 \), respectively. The resulting expressions are

\[ a_{i-1/2,L} = \frac{1}{60} \left( -3 \overline{a_{i-2}} + 27 \overline{a_{i-1}} + 47 \overline{a_i} - 13 \overline{a_{i+1}} + 2 \overline{a_{i+2}} \right), \]  

(A.3)

\[ a_{i+1/2,L} = \frac{1}{60} \left( 2 \overline{a_{i-2}} - 13 \overline{a_{i-1}} + 47 \overline{a_i} + 27 \overline{a_{i+1}} - 3 \overline{a_{i+2}} \right). \]  

(A.4)

The limiter we apply to passive scalars in the PQH method is the same as that in the PPH method, see above. The PQH method is exact wherever \( a(\xi) \) is locally a quartic polynomial, 5\textsuperscript{th}-order accurate for general but smooth functions \( a(\xi) \), and it requires three ghost cells at domain boundaries.

We have only performed a small number of tests with these two methods. Specifically, we ran simulations of the Kelvin-Helmholtz problem (Sect. 3.1) with the initial Mach number of \( M_0 \approx 10^{-3} \) and the HLLC flux function on grids of 64 × 32 and 128 × 64 cells. There were no overshoots in the passive tracer in the results of these tests.

Appendix B: 1D test cases

We use two simple 1D experiments – linear advection and the propagation of a linear sound wave – to compare the accuracy that the six reconstruction schemes can reach if not constrained by the 2\textsuperscript{nd}-order accuracy limit imposed by our multidimensional scheme.

B.1. Linear advection

The initial conditions in this experiment correspond to a right-going contact wave:

\[ \rho(x) = \gamma \left[ 1 + 0.01 \sin(2\pi x) \right], \]  

(B.1)

\[ u(x) = 0.1, \]  

(B.2)

\[ p(x) = 1. \]  

(B.3)

We consider the interval \( 0 \leq x \leq 1 \) with periodic boundary conditions and the equation of state of an ideal gas with the ratio of specific heats \( \gamma = 1.4 \). The average speed of sound is unity and it varies by 0.5% due to the density variation. The solution is sought at \( t = 10 \), when the sinusoid has been advected by one period and the analytic solution becomes identical to the initial condition. This makes the quantification of numerical errors trivial.

We use the same code for this experiment as we do for all the other experiments reported in this work. Since some of the schemes tested exceed 2\textsuperscript{nd} order of accuracy, cell averages cannot be approximated by sampling the initial condition at cell centers. We initialize the discrete density profile using the formula

\[ \rho_i = \gamma \left[ 1 + 0.01 \frac{\sin(\pi \Delta x i)}{\pi \Delta x} \sin(2\pi x_i) \right], \]  

(B.4)

where \( \Delta x \) is the constant grid spacing and \( x_i \) the location of \( i \)-th cell’s center. This formula is easily obtained by analytically averaging Eq. (B.1) in the interval \( x_i - \Delta x/2 \leq x \leq x_i + \Delta x/2 \).

We suppress time-stepping errors by using the RK3 scheme with CFL = 0.01. Because the Mach number of the flow is 0.1, this means that the wave is advected by only \( 10^{-3} \Delta x \) during every 3\textsuperscript{rd}-order-accurate time step. We use the LHLLC flux function for this experiment.

The L\textsuperscript{1} errors we obtain on grids with different numbers of cells \( N \), shown in Fig. B.1. We test the LIN, LIN+VL, PAR, and PPM84 schemes on grids with 8 to 512 cells. We are forced to stop at grids of 64 cells for the most accurate schemes PPM08 and PSH, because the errors rapidly become dominated by the finite precision of floating-point arithmetic. The asymptotic orders of accuracy based on L\textsuperscript{1} errors measured on the three finest grids available are reported in Table 1. Both LIN and LIN+VL reach the expected 2\textsuperscript{nd} order of accuracy. The absolute errors produced by LIN+VL are somewhat larger as compared with LIN because of the presence of the van Leer limiter. The PPM84 scheme is formally 4\textsuperscript{th}-order accurate but that only holds for monotonic solutions. PPM84 contains a limiter that flattens the slope at local extrema and that reduces the scheme’s asymptotic order to 2.3 in this test case. We have experimentally turned off both limiters present in PPM84, which increased the asymptotic order of accuracy to 4.0, as expected. The PAR, PPM08, and PSH methods reach the asymptotic orders of 3.0, 6.0, and 7.0, also matching theoretical expectations.

B.2. Sound wave

The initial conditions in this experiment correspond to a right-going sound wave:

\[ \rho(x) = \rho_0 \left[ 1 + \gamma^{-1} \Psi(x) \right], \]  

(B.5)

\[ u(x) = \gamma^{-1} c_0 \Psi(x), \]  

(B.6)

\[ p(x) = p_0 \left[ 1 + \Psi(x) \right], \]  

(B.7)

where

\[ \Psi(x) = \Psi_0 \sin(2\pi x). \]  

(B.8)
We use $\rho_0 = \gamma$, $p_0 = 1$ and the equation of state of an ideal gas with the ratio of specific heats $\gamma = 1.4$, so that the unperturbed speed of sound is $c_0 = 1$. We set the amplitude $\Psi_0 = 10^{-10}$ to suppress nonlinear effects. This allows us to avoid accuracy constraints imposed by our $2^{\text{nd}}$-order-accurate transformations between primitive and conserved variables. Round-off errors are suppressed by using 128-bit floating point numbers. In a way analogous to Eq. (B.4), we initialize the discrete wave such that the cell average of $\Psi(x)$ in cell $i$ is

$$\Psi_i = \frac{\sin (\pi x_i)}{\pi x_i} \Psi(x_i).$$

The simulations are stopped at $t = 1$, when the sound wave has propagated by a single wavelength around the periodic domain $0 \leq x \leq 1$. Neglecting nonlinear effects, which are of order $\Psi_0 = 10^{-20}$, the evolved solution is expected to be identical to the initial condition. Timestepping errors are suppressed by using the RK3 scheme with CFL = 0.001, so the wave moves by only $10^{-3} \Delta x$ per time step. We use the LHLLC flux function for this experiment.

We show $L^1$ errors in the velocity $u$ on grids with 8 to 512 cells in Fig. B.2. The simulation series with the most accurate PPM08 and PSH schemes are stopped at grids of 64 cells, because the absolute errors rapidly approach $10^{-20}$, that is the magnitude of the residual nonlinear effects. The asymptotic orders of accuracy, as defined by the $L^1$ errors measured on the three finest grids available, are 2.0, 3.0, 6.0, and 7.0 for the LIN, PAR, PPM08, and PSH methods, respectively, matching theoretical expectations. The LIN+VL and PPM84 methods only reach the order of 1.6, which seems to be caused by the combination of the low-Mach flux function LHLLC with the limiters contained in these two methods. We have checked that both methods reach the order of 2.0 with the RUSANOV and HLLC flux functions (PPM84 flattens the slope at local extrema, which reduces its order, see also Sect. B.1).

The $L^1$ errors discussed so far do not distinguish between amplitude and phase errors. We quantify amplitude errors by computing the relative loss of the total kinetic energy from $t = 0$ to $t = 1$, see Fig. B.3. Some kinetic energy is lost in all of the simulations, which is a sign of stability. The amount of energy dissipated decreases with the $3^{\text{rd}}$ power of the grid spacing for LIN, LIN+VL, PAR, and PPM84. It is not immediately clear why this is the case given that the methods have different orders of accuracy, but we did not investigate this further. In case of PSH, the dissipation rate decreases with the $7^{\text{th}}$ power of the grid spacing, matching the method’s order of accuracy. PPM08 is a special case – the method preserves extrema and sinusoids resolved by 16 or more cells per wavelength turn out to be smooth enough not to trigger any of the method’s limiters. PPM08 reduces to a simple interpolation function in this special case, eliminating jumps at all cell interfaces, see Sect. 2.3.5. This, in turn, eliminates all explicit dissipative terms in the flux function. The dissipation rate drops by many orders of magnitude and timestepping errors start to dominate.

We refer to Appendix A for a discussion of the relative loss of the total kinetic energy.

The simulations are stopped at $t = 1$, when the sound wave has propagated by a single wavelength around the periodic domain $0 \leq x \leq 1$. Neglecting nonlinear effects, which are of order $\Psi_0 = 10^{-20}$, the evolved solution is expected to be identical to the initial condition. Timestepping errors are suppressed by using the RK3 scheme with CFL = 0.001, so the wave moves by only $10^{-3} \Delta x$ per time step. We use the LHLLC flux function for this experiment.

We show $L^1$ errors in the velocity $u$ on grids with 8 to 512 cells in Fig. B.2. The simulation series with the most accurate PPM08 and PSH schemes are stopped at grids of 64 cells, because the absolute errors rapidly approach $10^{-20}$, that is the magnitude of the residual nonlinear effects. The asymptotic orders of accuracy, as defined by the $L^1$ errors measured on the three finest grids available, are 2.0, 3.0, 6.0, and 7.0 for the LIN, PAR, PPM08, and PSH methods, respectively, matching theoretical expectations. The LIN+VL and PPM84 methods only reach the order of 1.6, which seems to be caused by the combination of the low-Mach flux function LHLLC with the limiters contained in these two methods. We have checked that both methods reach the order of 2.0 with the RUSANOV and HLLC flux functions (PPM84 flattens the slope at local extrema, which reduces its order, see also Sect. B.1).

The $L^1$ errors discussed so far do not distinguish between amplitude and phase errors. We quantify amplitude errors by computing the relative loss of the total kinetic energy from $t = 0$ to $t = 1$, see Fig. B.3. Some kinetic energy is lost in all of the simulations, which is a sign of stability. The amount of energy dissipated decreases with the $3^{\text{rd}}$ power of the grid spacing for LIN, LIN+VL, PAR, and PPM84. It is not immediately clear why this is the case given that the methods have different orders of accuracy, but we did not investigate this further. In case of PSH, the dissipation rate decreases with the $7^{\text{th}}$ power of the grid spacing, matching the method’s order of accuracy. PPM08 is a special case – the method preserves extrema and sinusoids resolved by 16 or more cells per wavelength turn out to be smooth enough not to trigger any of the method’s limiters. PPM08 reduces to a simple interpolation function in this special case, eliminating jumps at all cell interfaces, see Sect. 2.3.5. This, in turn, eliminates all explicit dissipative terms in the flux function. The dissipation rate drops by many orders of magnitude and timestepping errors start to dominate.

We refer to Appendix A for a discussion of the relative loss of the total kinetic energy.
Appendix C: Sound wave generation with a compact third order WENO scheme

The results presented in Sects. 3.1 and 3.2 reveal that nonlinear, switching reconstruction schemes (such as LIN+VL, PPM84, and PPM08) generate excess acoustic power which is not present in simulations run with unlimited (linear) reconstruction methods. The question naturally arises whether nonlinear or non-switching schemes (e.g., WENO) also generate artificial sound waves in simulations of low-Mach-number flows. In this Appendix we try to answer this question using the compact third order (CTO) WENO scheme of Kolb (2014). This scheme constructs the quantity $a$ within the cell $i$ using a weighted sum of three polynomials, $a_{R,C}(\zeta)$. The ultimate expression for the underlying function $a(\zeta)$ reads

$$a(\zeta) = w^{L}_m a_{L}(\zeta) + w^{C}_m a_{C}(\zeta) + w^{R}_m a_{R}(\zeta). \quad (C.1)$$

The weighting coefficients are computed as

$$w_m = \frac{a_m}{\sum_{k \in [L,C,R]} a_k}, \quad (C.2)$$

where

$$a_m = \frac{m}{(\epsilon_{WENO} + IS_m)^{p}}, \quad (C.3)$$

$m \in [L,C,R]$, $C_k = \epsilon_k = 0.25$, $C_C = 0.5$, and IS$_m$ are the smoothness indicators of the scheme$^{21}$,

$$IS_L = (\bar{a}_i - \bar{a}_{i-1})^2, \quad (C.4)$$

$$IS_R = (\bar{a}_{i+1} - \bar{a}_i)^2, \quad (C.5)$$

$$IS_C = \frac{13}{12} L \epsilon \left(2 \bar{a}_i - 2 \bar{a}_{i+1} + \bar{a}_{i-1}\right)^2 + \frac{1}{4} (\bar{a}_{i+1} - \bar{a}_{i-1})^2. \quad (C.6)$$

In Eq. (C.3), we set $p = 2$. The functions $a(\zeta)_{L,R}$ are one-sided linear reconstructions

$$a_L(\zeta) = \bar{a}_i + (\bar{a}_i - \bar{a}_{i-1})(\zeta - \xi), \quad (C.7)$$

$$a_R(\zeta) = \bar{a}_{i+1} + (\bar{a}_{i+1} - \bar{a}_i)(\zeta - \xi), \quad (C.8)$$

whereas $a_C(\zeta)$ is defined such that

$$a_{C}(\zeta) = c_L a_L(\zeta) + c_C a_C(\zeta) + c_R a_R(\zeta) \quad (C.9)$$

is the unique parabola that conserves the cell volume averages $\bar{a}_{i-1}$, $\bar{a}_i$, and $\bar{a}_{i+1}$ over cells $i-1$, $i$, and $i+1$, respectively. This constraint implies

$$a_C(\zeta) = a_i + \frac{\partial a}{\partial \zeta} \bigg|_{\zeta = \xi} (\zeta - \xi) + \frac{1}{2} \frac{\partial^2 a}{\partial \zeta^2} \bigg|_{\zeta = \xi} (\zeta - \xi)^2, \quad (C.10)$$

with

$$a_i = \bar{a}_i - \frac{1}{24} (2 \bar{a}_{i+1} - 2 \bar{a}_i + \bar{a}_{i-1}), \quad (C.11)$$

$$\frac{\partial a}{\partial \zeta} \bigg|_{\zeta = \xi} = \frac{\bar{a}_{i+1} - \bar{a}_{i-1}}{2}, \quad (C.12)$$

$$\frac{\partial^2 a}{\partial \zeta^2} \bigg|_{\zeta = \xi} = \frac{\bar{a}_{i+1} - 2 \bar{a}_i + \bar{a}_{i-1}}{2}. \quad (C.13)$$

The smoothness indicators IS$_{L,R,C}$ are such that the smoothest among the polynomials $a(\zeta)_{L,R,C}$ has the largest weight in

---

21 Here we assume that the reconstructed variable $a$ is dimensionless.

---

Appendix D: Additional plots for the Kelvin–Helmholtz problem

G. Leidi et al.: Performance of high-order Godunov methods
Fig. C.1. Frequency power spectrum of the vertical velocity component $v$ extracted in the middle of the stable layer at $(x, y, z) = (-0.2, 2.5, -0.2)$ over the time series $t \in (10\tau_{\text{conv}}, 32\tau_{\text{conv}})$ in the problem of turbulent convection and wave excitation described in Sect. 3.2. These results are obtained on a grid with $128^3$ cells, using the LHLLC Riemann solver and the CTO-WENO scheme of Kolb (2014) for different values of the parameter $\epsilon_{\text{WENO}}$ occurring in the smoothness indicators of the scheme (see Sect. C), ranging from $10^{-12}$ to $10^2$. As a reference, power spectra obtained with the numerical options LHLLC + PPM84 and LHLLC + LIN + VL are also shown. The convective turnover frequency ($\omega_{\text{conv}} = 2\pi/\tau_{\text{conv}}$), the Brunt-Väisälä frequency at the location of the point probe ($N_{\text{BV}}$), and the frequency of the fundamental oscillation mode of the cavity ($\omega_0 = 1.1$) are represented by the black dashed-dotted, dashed, and dotted lines, respectively.
Fig. D.1. As Fig. 4 but with the initial Mach number $M_0 = 10^{-1}$.
Fig. D.2. As Fig. 4 but with the initial Mach number $M_0 = 10^{-3}$.
Fig. D.3. As Fig. 5 but with the initial Mach number $M_0 = 10^{-1}$. 
As Fig. 5 but with the initial Mach number $M_0 = 10^{-3}$.
Fig. D.5. As Fig. 6 but with the initial Mach number $M_0 = 10^{-1}$. 
Fig. D.6. As Fig. 6 but with the initial Mach number $M_0 = 10^{-3}$.
Fig. D.7. As Fig. 4 but showing simulations computed on the 4096 × 2048 grid.
Fig. D.8. As Fig. 6 but showing simulations computed on the $4096 \times 2048$ grid.