

thornado-hydro: towards discontinuous Galerkin methods for supernova hydrodynamics*

Eirik Endeve^{1,2,6}, Jesse Buffaloe², Samuel J Dunham³,
Nick Roberts², Kristopher Andrew^{4,1}, Brandon Barker²,
David Pochik², Juliana Pulsinelli^{5,7} and Anthony Mezzacappa^{2,6}

¹Computer Science and Mathematics Division, Oak Ridge National Laboratory, TN 37831

²Department of Physics and Astronomy, University of Tennessee Knoxville, TN 37996

³Department of Astronomy, Vanderbilt University, TN 37212

⁴Department of Physics and Astronomy, University of Kentucky, Lexington, KY 40506

⁵Princeton University, NJ 08544

⁶Joint Institute for Computational Sciences, Oak Ridge National Laboratory, TN 37831

⁷Physics Division, Oak Ridge National Laboratory, TN 37831

E-mail: endevee@ornl.gov

Abstract. The toolkit for high-order neutrino-radiation hydrodynamics (**thornado**) is being developed for simulations of core-collapse supernovae (CCSNe) and related problems. Current capabilities in **thornado** include solvers for the Euler equations — in non-relativistic and special relativistic limits — and the two-moment model of neutrino transport. The spatial discretization in **thornado** is based on the discontinuous Galerkin (DG) method, which is receiving increased attention from the computational astrophysics community. In this paper, we provide an overview of the numerical methods for the Euler equations in **thornado**, and present some encouraging preliminary numerical results from a set of basic tests in one and two spatial dimensions.

1. Introduction

Along with solvers for neutrino transport and gravity, the hydrodynamics solver constitutes a major component of core-collapse supernova (CCSN) simulations. The dynamics of the bounce shock and associated fluid instabilities, which drive turbulent flows and help shape the explosion, must be faithfully captured (see, e.g., [23], for a recent review). Essentially, the stellar interior is modeled as a perfect fluid (thermal conduction and viscosity are not explicitly included) using the Euler equations, but the models are extended to accommodate a nuclear equation of state.

The discontinuous Galerkin (DG) method [7, 14] appears as an appealing choice to model fluid flows in CCSNe. DG methods combine elements of spectral and finite volume methods, and achieve high-order accuracy on a compact stencil. Data is only communicated with nearest

*This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<http://energy.gov/downloads/doe-public-access-plan>).



neighbors, regardless of the formal order of accuracy, which leads to a high computation to communication ratio, and favorable parallel scalability on heterogeneous architectures has been demonstrated [17]. They can be used in combination with *hp*-adaptivity [26], where, in addition to grid refinement with AMR, the local polynomial degree can be chosen differently, and independently, in different cells. DG methods can easily be applied to problems involving curvilinear coordinates, which is beneficial in numerical relativity [36]. Currently, most hydrodynamics solvers in CCSN codes are based on finite volume methods. Exploring the utility and performance of DG methods to model CCSNe seems like a worthy exercise by itself.

The toolkit for **high-order neutrino-radiation hydrodynamics** (**thornado**) is being developed to simulate neutrino-radiation hydrodynamics in CCSNe and related applications in nuclear astrophysics. The spatial discretization of solvers for hyperbolic partial differential equations in **thornado** is based on the DG method, while we foresee using a combination of spectral and continuous finite element methods for solving elliptic equations; e.g., for Newtonian gravity or relativistic gravity employing the conformal flatness approximation [39]. With these approaches, we employ high-order discretization techniques based on a common mathematical framework in all the major model components. Whether the high-order approach will improve accuracy and efficiency of CCSN models remains to be demonstrated. However, in this paper, we provide an initial description and encouraging demonstration of the DG method implemented in **thornado** to solve the Euler equations. We focus on basic tests in one and two spatial dimensions and focus, to some extent, on the implementation of limiters in **thornado**. We note that others (e.g., [25, 28, 43, 10]) have also investigated the DG method in the context of astrophysical applications and reported encouraging results. The DG limiting techniques implemented in **thornado** are similar to those used by [28], but is supplemented with a troubled-cell indicator [12] to prevent excessive limiting (e.g., around smooth extrema), and differs from more recent approaches in the literature (e.g., [15, 9, 43, 35]), which combine the sub-cell resolution of the DG method with high-order finite-volume techniques to improve the performance of the DG limiting procedure. While the results with current limiters are encouraging, these latter methods appear as attractive options for future implementation in **thornado**.

2. The discontinuous Galerkin method

Excellent books and review articles on the DG method are available (see, e.g., [6, 7, 14, 33] and references therein), and we will not go into too much detail here. However, we review some key concepts to introduce notation, and emphasize specific choices for our implementation in **thornado**. To this end, we consider a system of conservation laws with sources of the form

$$\partial_t(\sqrt{\gamma}\mathbf{U}) + \sum_{i=1}^d \partial_i(\sqrt{\gamma}\mathbf{F}^i(\mathbf{U})) = \sqrt{\gamma}\mathbf{S}(\mathbf{U}), \quad (1)$$

where $\mathbf{U}$ is the evolved state vector, $\mathbf{F}^i$ are the fluxes, and $\mathbf{S}$ is the source vector. We use a formulation of the equations sufficiently general to accommodate curvilinear spatial coordinates encoded in the metric γ_{ij} , giving the squared line element $ds^2 = \gamma_{ij}dx^i dx^j$, whose determinant is γ . Henceforth, we will assume that the spatial metric is time-independent. To solve Eq. (1), the computational domain $D \subset \mathbb{R}^d$ is divided into a disjoint union $\mathcal{T}$ of open elements $\mathbf{K}$, so that $D = \cup_{\mathbf{K} \in \mathcal{T}} \mathbf{K}$. We require that each element is a box in the logical coordinates; i.e.,

$$\mathbf{K} = \{ \mathbf{x} : x^i \in K^i := (x_L^i, x_H^i), i = 1, \dots, d \}, \quad (2)$$

with the surface elements denoted $\partial\mathbf{K}^i = \times_{j \neq i} K^j$. We let $V_{\mathbf{K}}$ denote the proper element volume

$$V_{\mathbf{K}} = \int_{\mathbf{K}} dV, \quad \text{where} \quad dV = \sqrt{\gamma} \prod_{i=1}^d dx^i. \quad (3)$$

We also define as a set $\mathbf{x} = \{\tilde{\mathbf{x}}^i, x^i\}$ and $\Delta x^i = x_{\text{H}}^i - x_{\text{L}}^i$.

We let the approximation space for the DG method, $\mathbb{V}^k$, be constructed from the tensor product of one-dimensional polynomials of maximal degree k . Note that functions in $\mathbb{V}^k$ can be discontinuous across element interfaces. The semi-discrete DG problem is to find $\mathbf{U}_h \in \mathbb{V}^k$, which approximates $\mathbf{U}$ in Eq. (1), such that for all $v \in \mathbb{V}^k$ and all $\mathbf{K} \in \mathcal{T}$

$$\begin{aligned} \partial_t \int_{\mathbf{K}} \mathbf{U}_h v dV + \sum_{i=1}^d \int_{\partial \mathbf{K}^i} (\sqrt{\gamma} \hat{\mathbf{F}}^i(\mathbf{U}_h) v|_{x_{\text{H}}^i} - \sqrt{\gamma} \hat{\mathbf{F}}^i(\mathbf{U}_h) v|_{x_{\text{L}}^i}) d\tilde{\mathbf{x}}^i \\ - \sum_{i=1}^d \int_{\mathbf{K}} \mathbf{F}^i(\mathbf{U}_h) \partial_i v dV = \int_{\mathbf{K}} \mathbf{S}(\mathbf{U}_h) v dV. \end{aligned} \quad (4)$$

In Eq. (4), $\hat{\mathbf{F}}^i(\mathbf{U}_h)$ is a numerical flux approximating the flux on the i th surface of $\mathbf{K}$. The numerical flux function $\mathbf{f}^i$ is evaluated using values from both sides of an element interface; i.e.,

$$\hat{\mathbf{F}}^i(\mathbf{U}_h) = \mathbf{f}^i(\mathbf{U}_h(x^{i,-}, \tilde{\mathbf{x}}^i), \mathbf{U}_h(x^{i,+}, \tilde{\mathbf{x}}^i)), \quad (5)$$

where superscripts $-/+$, e.g. in the arguments of $\mathbf{U}_h$, indicate that the function is evaluated to the immediate left/right of x^i . We use the Harten-Lax-van Leer (HLL) flux [13] or the HLLC flux [37, 22] for all the numerical experiments presented in Section 3.

We provide further details on the DG method to arrive at the equations that are actually evolved in **thornado**. We start by introducing some notation, defining the polynomial expansion for $\mathbf{U}_h$ and the quadrature rules used to evaluate the integrals in Eq. (4). Then we provide explicit expressions for each of the terms in Eq. (4). For simplicity we consider one spatial dimension ($d = 1$) and drop indices denoting the spatial dimension. In each element K , we use a nodal representation of the conserved variables $\mathbf{U}$; i.e.,

$$\mathbf{U}(x, t) \approx \mathbf{U}_h(x, t) = \sum_{i=1}^N \mathbf{U}_i(t) \ell_i(x), \quad \text{where} \quad \ell_i(\eta) = \prod_{\substack{j=1 \\ j \neq i}}^N \frac{\eta - \eta_j}{\eta_i - \eta_j} \quad (6)$$

are Lagrange polynomials defined on the reference element $I = \{\eta : \eta \in (-0.5, 0.5)\}$, and constructed to interpolate the node set $S_N = \{\eta_i\}_{i=1}^N \subset I$. The spatial coordinate x and the reference coordinate η are related by the mapping $x(\eta) = x_{\text{L}} + (0.5 + \eta) \Delta x$. Then, for any $\eta_j \in S_N$, $\ell_i(\eta_j) = \delta_{ij}$, so that $\mathbf{U}_h(x(\eta_j), t) = \mathbf{U}_j(t)$. We introduce numerical quadratures to evaluate the integrals in Eq. (4). First we define the M -point quadrature $Q_M : C^0(I) \rightarrow \mathbb{R}$ with abscissas $\hat{S}_M = \{\eta_q\}_{q=1}^M$ and weights $\{w_q\}_{q=1}^M$, normalized such that $\sum_{q=1}^M w_q = 1$. The M -point Legendre-Gauss quadrature, which we use, integrates polynomials of degree $\leq 2M - 1$ exactly. Then, if $P_h(x)$ is such a polynomial, we have

$$\frac{1}{\Delta x} \int_K P_h(x) dx = \int_I P_h(\eta) d\eta = Q_M[P_h] \equiv \sum_{q=1}^M w_q P_h(\eta_q). \quad (7)$$

Note that the interpolation points S_N and the quadrature points $\hat{S}_M$ generally do not coincide. However, for the sake of computational efficiency we let $M = N$ and $S_N = \hat{S}_N$, which is a spectral-type nodal collocation DG approximation [2]. Inserting Eq. (6) into Eq. (4), letting $v(x) = \ell_k(x)$, and using the quadratures defined above, we obtain

$$\partial_t \int_K \mathbf{U}_h v dV \approx w_k \sqrt{\gamma_k} \partial_t \mathbf{U}_k \Delta x \quad (8)$$

for the time derivative term. For a general metric, the integral in Eq. (8) is approximate since we use the Legendre-Gauss quadrature rule with the nodal points given by the expansion in Eq. (6). However, since $\ell_i(\eta_k) = \delta_{ik}$, this leads to a diagonal mass matrix, and simplifies the implementation. Similarly, we obtain

$$\int_K \mathbf{S}(\mathbf{U}_h) v dV \approx w_k \sqrt{\gamma_k} \mathbf{S}(\mathbf{U}_k) \Delta x. \quad (9)$$

for the source term. Finally, the volume term (last term on the left-hand side of Eq. (4)) becomes

$$\int_K \mathbf{F}(\mathbf{U}_h) \frac{\partial v}{\partial x} dV \approx \sum_{q=1}^N w_q \sqrt{\gamma_q} \mathbf{F}(\mathbf{U}_q) \frac{\partial \ell_k}{\partial \eta}(\eta_q). \quad (10)$$

Using Eqs. (8)-(10) in Eq. (4) results in the semi-discrete form

$$\frac{d\mathbf{U}_k}{dt} = -\frac{1}{w_k \sqrt{\gamma_k} \Delta x} \left\{ \left[\sqrt{\gamma} \hat{\mathbf{F}} \ell_k|_{x_H} - \sqrt{\gamma} \hat{\mathbf{F}} \ell_k|_{x_L} \right] - \sum_{q=1}^N w_q \sqrt{\gamma_q} \mathbf{F}(\mathbf{U}_q) \frac{\partial \ell_k}{\partial \eta}(\eta_q) \right\} + \mathbf{S}(\mathbf{U}_k). \quad (11)$$

Eq. (11) comprises a system of ordinary differential equations (ODEs), which are integrated in time with an ODE solver. In Section 3 we use the optimal third-order strong stability-preserving Runge-Kutta (SSP-RK3) method from [30]. We assume that the source term $\mathbf{S}(\mathbf{U}_k)$ is non-stiff so that the equations can be integrated efficiently with explicit methods. In future applications of the DG method to model CCSNe we plan to use a combination of implicit and explicit methods (e.g., IMEX methods [1]) to stably integrate stiff sources due to neutrino-matter interactions.

Limiting of the polynomial representation $\mathbf{U}_h$, to reduce oscillations and prevent unphysical states, is a critical step in the DG algorithm. We use the TVD-type slope limiter based on limiting characteristic variables, discussed in [6]. To prevent excessive limiting (e.g., at smooth extrema), we use the troubled-cell indicator (TCI) of [12]:

$$I_K(G) = \frac{\sum_j |G_K - G_K^{(j)}|}{\max_j |G_K^{(j)}|}, \quad (12)$$

where $G \in \mathbf{G} \subseteq \mathbf{U}$. (Here, $\mathbf{G}$ consists of the first and last element of $\mathbf{U}$.) In Eq. (12), the sum in the numerator and the max in the denominator extend over the neighboring elements sharing a face with the target element $\mathbf{K}$. G_K is the cell average in $\mathbf{K}$, $G_K^{(j)}$ is the cell average computed by extrapolating the polynomial representation from the neighbor element $\mathbf{K}^{(j)}$ into $\mathbf{K}$, and $G_{K^{(j)}}^{(j)}$ is the cell average native to $\mathbf{K}^{(j)}$. An element is flagged for limiting if $I_K(G) > C_{\text{TCI}}$ for any $G \in \mathbf{G}$. To prevent negative mass density and pressure (and superluminal velocity for the relativistic Euler equations), we follow the approaches in [44, 40, 24].

To summarize the full RK-DG algorithm we let $\bar{\mathbf{U}}$ denote the global solution vector containing all the unknowns in D and $\bar{\mathbf{F}}$ denote the corresponding DG discretization from the right-hand side of Eq. (11), so that the global ODE system can be written as the $d_t \bar{\mathbf{U}} = \bar{\mathbf{F}}(\bar{\mathbf{U}})$. The TVD limiter and the ‘positivity’ limiter preventing unphysical states are denoted by $\Lambda^{\text{TVD}}\{\}$ and $\Lambda^{\text{Pos}}\{\}$, respectively (e.g., application of the TVD limiter is $\bar{\mathbf{U}} := \Lambda^{\text{TVD}}\{\bar{\mathbf{U}}\}$). The general s -stage Runge-Kutta time stepping algorithm, which includes application of limiters, can then be written as [7]:

1. Set $\bar{\mathbf{U}}^{(0)} = \bar{\mathbf{U}}^n$,

2. For $i = 1, \dots, s$ compute:

$$\bar{\mathbf{U}}^{(i)} = \Lambda^{\text{Pos}} \left\{ \Lambda^{\text{TVD}} \left\{ \sum_{j=0}^{i-1} \left(\alpha_{ij} \bar{\mathbf{U}}^{(j)} + \beta_{ij} \Delta t \bar{\mathbf{F}}(\bar{\mathbf{U}}^{(j)}) \right) \right\} \right\}, \quad (13)$$

3. Set $\bar{\mathbf{U}}^{n+1} = \bar{\mathbf{U}}^{(s)}$.

The coefficients α_{ij} and β_{ij} for SSP-RK3 are given in Table 2.1 in [7]. In Eq. (13), for each stage, the TVD limiter is applied first (for elements where $I_{\mathbf{K}} > C_{\text{TCL}}$), followed by the positivity limiter. The algorithm is subject to two time step restrictions, the stability condition Δt_{TVD} and the positivity condition Δt_{Pos} , and we take $\Delta t = \min_D(\Delta t_{\text{TVD}}, \Delta t_{\text{Pos}})$. Let $(h/|\lambda|) = \min_{i \in \{1, \dots, d\}}(\Delta x^i/|\lambda^i|)$, where $|\lambda^i|$ is the largest absolute eigenvalue of the flux Jacobian. Then, assuming Cartesian coordinates (e.g., [7, 44]):

$$\Delta t_{\text{TVD}} \leq \frac{1}{d} \frac{(h/|\lambda|)}{(2k+1)} \quad \text{and} \quad \Delta t_{\text{Pos}} \leq \frac{\hat{w}_1(h/|\lambda|)}{d}, \quad (14)$$

where $\hat{w}_1$ is the first quadrature weight of the Gauss-Lobatto quadrature that integrates the cell average of $\mathbf{U}_h$ exactly. For $d = k = 2$, the effective Courant-Friedrichs-Lewy (CFL) factor is 0.1 for Δt_{TVD} , while it is 1/12 for Δt_{Pos} . The positivity limiter checks for unphysical states in all the quadrature points involved in the DG discretization. If any point in an element is flagged as unphysical, the polynomial representation $\mathbf{U}_h$ is damped towards its cell average $\mathbf{U}_{\mathbf{K}}$, which is guaranteed to be physical by the Δt_{Pos} restriction. Note that Δt_{Pos} is a sufficient condition. In practice, the time step can often be set larger than Δt_{Pos} .

3. Preliminary numerical results

In this section we show preliminary numerical results obtained with the DG method implemented in **thornado** to solve the non-relativistic and special relativistic Euler equations. We use the ideal gas equation of state where the pressure p is related to the internal energy density e by $p = (\Gamma - 1)e$, and where Γ is the (constant) adiabatic index. Let the d -dimensional computational domain be $D = \times_{i=1}^d [x_m^i, x_M^i]$, where x_m^i and x_M^i denote the coordinates of the inner and outer boundary in the i th coordinate direction, respectively. To limit the scope of this initial study, we use polynomials of degree $k = 2$ (except in Section 3.1.4, where we also use $k = 1$ and $k = 3$) combined with SSP-RK3 time stepping, and, unless stated otherwise, run all the tests with a CFL factor of $C = 0.1$. We use the HLL Riemann solver when solving the non-relativistic Euler equations and the HLLC Riemann solver when solving the special relativistic Euler equations. The purpose of the tests is to gauge the performance of the DG implementation on a set of benchmarks as an initial measure of its suitability for future CCSN simulations.

3.1. Non-relativistic (NR) hydrodynamics

For the NR Euler equations, the state, flux, and source vectors in Eq. (1) are

$$\mathbf{U} = (\rho, \rho u_j, E)^T, \quad \mathbf{F}^i = (\rho u^i, \Pi_j^i, (E + p)u^i)^T, \quad \text{and} \quad \mathbf{S} = (0, \frac{1}{2}\Pi^{ik}\partial_j\gamma_{ik} - \rho\partial_j\Phi, -\rho u^i\partial_i\Phi)^T, \quad (15)$$

where ρ and u^i are the mass density and components of the fluid three-velocity, respectively. The stress tensor is $\Pi_j^i = \rho u^i u_j + p\delta_j^i$, and $E = e + \frac{1}{2}\rho u_i u^i$ is the fluid (internal plus kinetic) energy density. The sources are due to curvilinear coordinates and Newtonian gravity.

3.1.1. Sod shock tube The first test is the classic Riemann problem due to Sod [34], computed using Cartesian coordinates. The one-dimensional computational domain is $D = [0, 1]$, and a discontinuity is initially located at $x^1 = 0.5$, separating the left and right states

$$\mathbf{U}_L = (1.0, \mathbf{0.0}, 2.5)^T \quad \text{and} \quad \mathbf{U}_R = (0.125, \mathbf{0.0}, 0.25)^T,$$

where the adiabatic index is $\Gamma = 1.4$. The test is run with 100 elements until $t = 0.2$, using $C_{\text{TCI}} = 0.03$. Results are plotted in the left panels of Figure 1. The DG method captures the main features of the solution well without introducing noticeable oscillations near the discontinuities (upper left panel). In the lower left panel we plot locations in the xt -plane of elements flagged for limiting. Limiting is mainly triggered by the contact discontinuity and the shock.

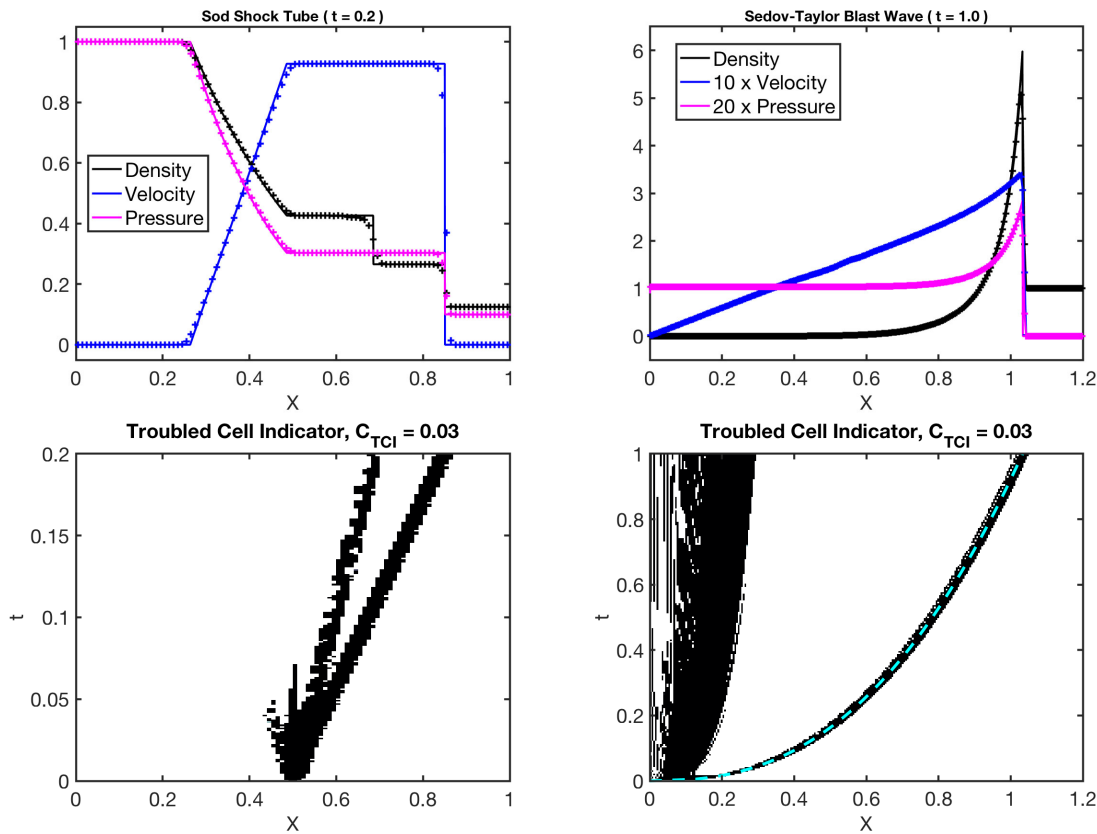


Figure 1. *Upper left panel:* numerical solution of Sod's problem using 100 elements (plusses) compared with the exact solution (solid lines; from [38]). *Upper right panel:* numerical solution of the Sedov-Taylor blast wave using 256 elements (plusses) against the exact solution (solid lines; from [16]). In the lower panels we plot locations in the xt -plane of elements flagged for limiting ($I_K > C_{\text{TCI}}$) by the Sod (left) and Sedov (right) problems.

3.1.2. Sedov-Taylor blast wave This test, detailed in [29] (see also §99 in [18]), is computed in spherical polar coordinates with the assumption of spherical symmetry.² The computational domain is $D = [0, 1.2]$, and the initial condition consists of a fluid at rest with density $\rho = 1$. The adiabatic index is $\Gamma = 1.4$, and an amount of thermal energy equal to 1 is released in the

² In this paper, all tests in spherical polar coordinates are computed with the assumption of spherical symmetry.

innermost element. We use 256 elements and run until $t = 1$, using $C_{\text{TCI}} = 0.03$. Again, the DG method captures the characteristic of the exact solution (cf. upper right panel of Figure 1). The maximum density in the shock is somewhat lower for the numerical solution than the exact value (about 5 versus 6). In the lower right panel we map locations of elements flagged for limiting in the xt -plane. The troubled-cell indicator tracks the shock well (the analytic shock trajectory is given by the dashed cyan line), but also flags elements at smaller radii with low density.

3.1.3. Shu-Osher shock tube This test from [31] is suitable for measuring the amount of dissipation in a numerical scheme. The one-dimensional computational domain is $D = [-5, 5]$, and a discontinuity is placed at $x^1 = -4$, separating the states

$$\mathbf{U}_L = (3.857143, 10.14185, 0.0, 0.0, 39.16667)^T \quad \text{and} \quad \mathbf{U}_R = (1 + 0.2 \times \sin(5x^1), \mathbf{0.0}, 2.5)^T,$$

where the adiabatic index is $\Gamma = 1.4$. The density variations ahead of the Mach=3 shock are compressed through the shock and result in higher-frequency variations downstream, which can be difficult to capture with an excessively dissipative scheme.

We run this test to $t = 1.8$ using 200 elements. In the two upper panels of Figure 2 we plot the density for various values of the troubled-cell indicator threshold C_{TCI} ; 0.0 (full limiting, green), 0.03 (magenta), 0.3 (red), and 3.0 (blue). Larger C_{TCI} implies less limiting. The results are compared with a reference solution computed with 2048 elements. With full limiting, the scheme is unable to capture the density variations behind the shock (cf. upper right panel in Figure 2). This is expected since the TVD-type limiter clips extrema. The results improve when the troubled-cell indicator is used. When $C_{\text{TCI}} = 0.03$, the variations are better resolved, but the amplitudes are much reduced relative to the reference solution. With $C_{\text{TCI}} = 3.0$, the variations are well resolved and the amplitudes are comparable to the reference. In the lower panels of Figure 2 we plot locations in the xt -plane of elements flagged for limiting by the troubled-cell indicator: $C_{\text{TCI}} = 0.03$ (left), $C_{\text{TCI}} = 0.3$ (middle), and $C_{\text{TCI}} = 3.0$ (right). With $C_{\text{TCI}} = 0.03$, limiting is triggered in a wide region around the shock and around the peaks of the three leftmost density variations in the upper left panel in Figure 2. With $C_{\text{TCI}} = 3.0$, limiting is only triggered in the shock.

3.1.4. Isentropic vortex This test from [32] is included to verify the accuracy of the DG method implemented in `thornado`. The periodic computational domain is given by $D = [-5, 5] \times [-5, 5]$. The initial density and pressure are given by $\rho_0 = \left(1 - \frac{(\Gamma-1)\beta^2}{(8\pi\Gamma)} \exp(1-r^2)\right)^{1/(\Gamma-1)}$ and $P_0 = \rho^\Gamma$, where $r = \sqrt{(x^1)^2 + (x^2)^2}$, while the velocity components are $v_0^1 = 1 - \frac{x^2\beta}{2\pi} \exp((1-r^2)/2)$ and $v_0^2 = 1 + \frac{x^1\beta}{2\pi} \exp((1-r^2)/2)$. The strength of the vortex is set to $\beta = 5$, and we use $\Gamma = 1.4$. We evolve this test until $t = t_f = 10$, when the vortex has returned to its initial position.

In Figure 3 we plot the density error measured in the infinity norm $L^\infty = \max_D |\rho(\mathbf{x}, t_f) - \rho_0(\mathbf{x})|$ versus number of cells the x^1 -dimension for various configurations. We ran with spatial resolutions of 25×25 , 50×50 , 100×100 , and 200×200 . In the first set of runs, we ran without any limiting and varied the polynomial degree k : $k = 1$ (solid red), $k = 2$ (solid black), and $k = 3$ (solid green). We observe the expected order of accuracy for $k = 1$ and $k = 2$ (second- and third-order accuracy, respectively; cf. dotted reference lines). For $k = 3$, the order of accuracy is close to fourth-order for the lower resolutions, but tends to third-order as the resolution increases, presumably due to the use of third-order time stepping. (Note that we ran the $k = 3$ case with a CFL factor of 1/14; cf. Eq. (14).) In the second set of runs, we kept $k = 2$ and ran with limiting turned on. We ran one set with $C_{\text{TCI}} = 0.03$ (dashed magenta) and one set with $C_{\text{TCI}} = 0.1$ (dashed cyan). When $C_{\text{TCI}} = 0.03$, the limiter is activated for the lowest resolution run and significantly reduces the accuracy of the method. For all the other runs, the limiter is not activated, and the results are identical to the runs with the limiter turned off.

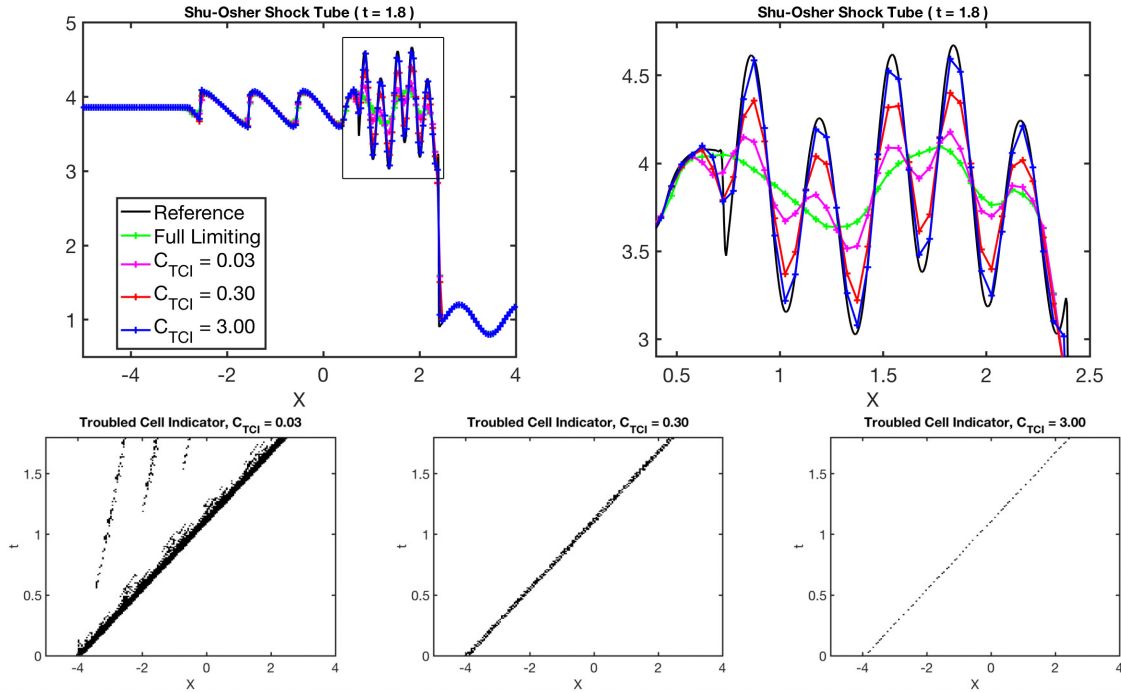


Figure 2. Results for the Shu-Osher shock tube problem computed with 200 elements. The upper panels show the mass density at $t = 1.8$ for various values of the troubled-cell indicator threshold C_{TCI} , compared with a high-resolution (2048 elements) reference solution. In the lower panels we show elements flagged for limiting in the xt -plane for various values of C_{TCI} .

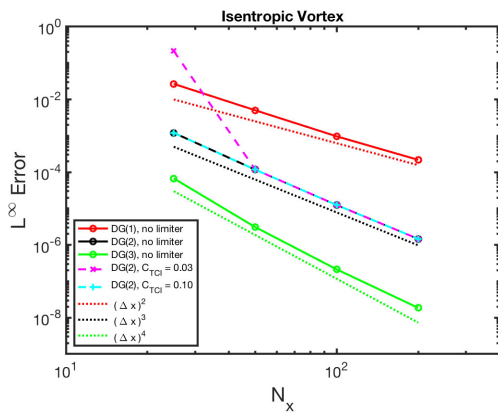


Figure 3. Convergence results for the Isentropic Vortex: L^∞ error norm versus number of elements in the x^1 -dimension.

secondary billows within the coiled-up interface separating dense (white) and less dense (black) fluids, while the model with no limiting has not. See Figure 5 for the spatial distribution of elements flagged for limiting at $t = 2.4$. When $t = 3.0$, these secondary instabilities appear to have largely destroyed the coil structure for the model with limiting, while the coils remain intact in the model without limiting (although secondary billows have started to form in this

3.1.5. Kelvin-Helmholtz (KH) instability This hydrodynamic instability is of significant astrophysical relevance — including during the explosion phase of CCSNe — and may occur in the interface separating fluids in relative motion. We compute this problem on the unit square $D = [0, 1] \times [0, 1]$, imposing periodic boundary conditions, and with initial conditions and single-mode perturbations taken from [21]. We use 256^2 elements and evolve until $t = 3.0$, well into the nonlinear regime. Results are displayed in Figure 4. We have computed two models, one with $C_{\text{TCI}} = 0.06$ (left panels) and one without limiting ($C_{\text{TCI}} \rightarrow \infty$; right panels).

Early on ($t = 1.5$), the results from the two runs are practically indistinguishable, show no sign of developing secondary billows, and agree visually with results presented in [21]. When $t = 2.4$, the run with $C_{\text{TCI}} = 0.06$ has started to develop

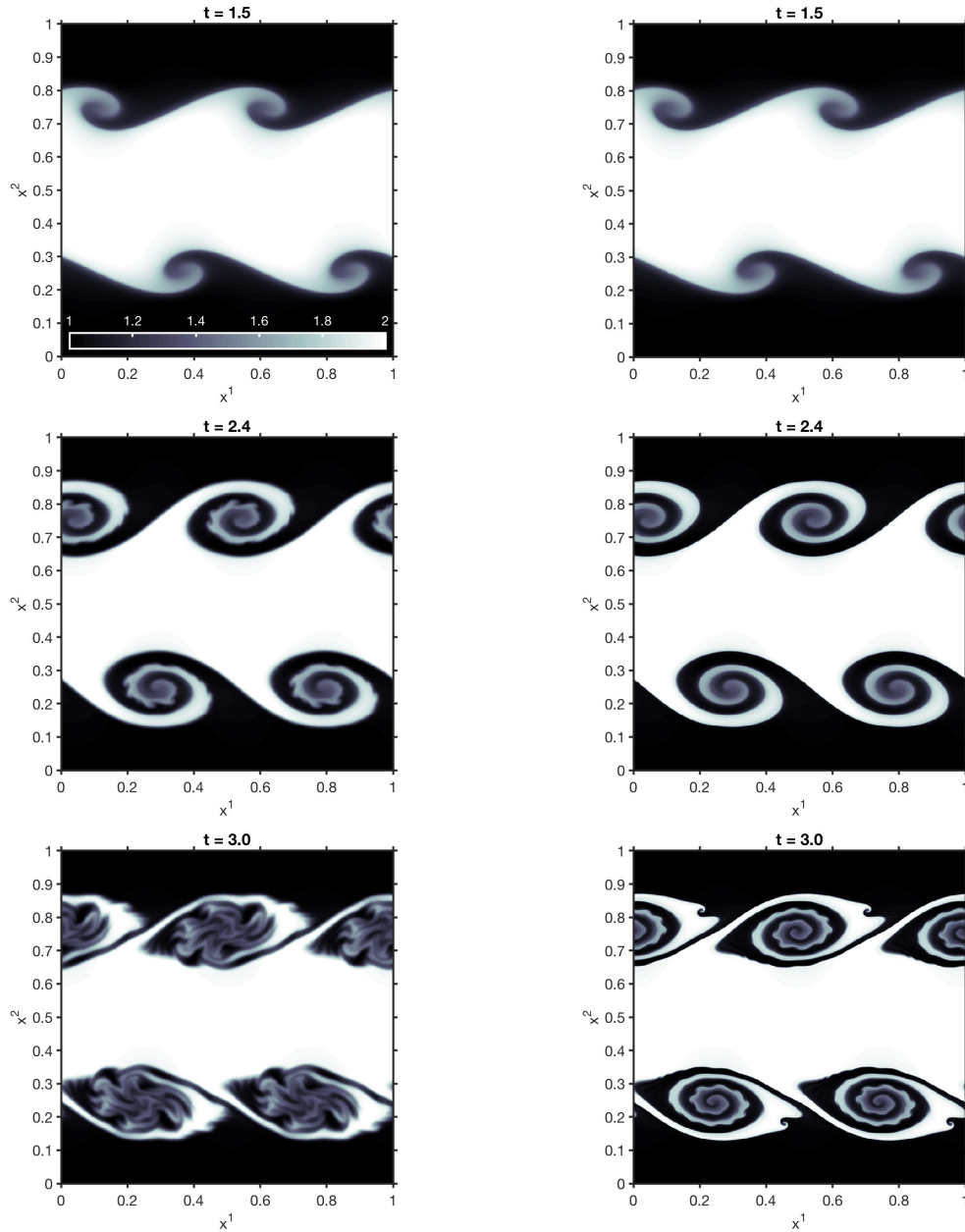


Figure 4. Mass density from the KH instability at various times, computed with 256^2 elements: $t = 1.5$ (top panels), $t = 2.4$ (middle panels), and $t = 3.0$ (bottom panels). Results for $C_{\text{TCI}} = 0.06$ and $C_{\text{TCI}} \rightarrow \infty$ (no limiting) are shown in the left and right panels, respectively.

model at this time). As discussed in more detail by [21], the development of the secondary billows may be an artifact of numerical perturbations and diffusion, which is delayed when the resolution is increased. Our results are consistent with this in the sense that the less diffusive model develops secondary billows later.

3.1.6. Liska-Wendroff implosion This test from [19] is computed on a 2D domain $D = [0, 0.3] \times [0, 0.3]$ with reflecting boundary conditions using 256^2 elements. Below the line

$x^2 = 0.15 - x^1$, we initially set $\mathbf{U} = (0.125, \mathbf{0.0}, 0.35)^T$, while $\mathbf{U} = (1.0, \mathbf{0.0}, 2.5)^T$ elsewhere.

Results for $t = 0.045$ and $t = 2.5$, with $C_{\text{TCl}} = 0.6$, are displayed in Figure 6 (left and right panels, respectively), which agree qualitatively with [19], who compared the results of eight different schemes for this problem. When $t = 0.045$, a shock wave propagates towards the origin of D , while a contact discontinuity is trailing the shock (cf. density contours), and a rarefaction wave is spreading in the opposite directions. When $t = 2.5$, a complex flow pattern with multiple shocks has emerged due to multiple reflections off the boundaries. The initial symmetry about the $x^1 = x^2$ diagonal is maintained, and the results from **thornado** agree qualitatively with the unsplit schemes in [19]. In the right panel of Figure 6, the characteristic “jet” has formed and propagates along the diagonal, similar to that displayed by CLAW and WENO in [19].

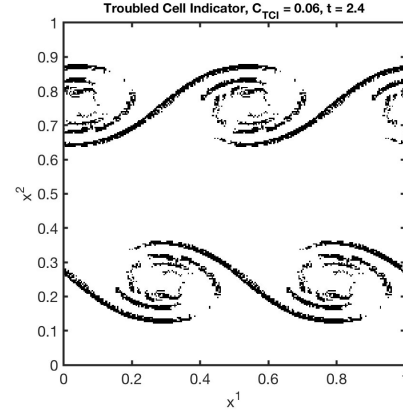


Figure 5. Distribution of elements flagged for limiting in the KH test at $t = 2.4$.

3.1.7. Liska-Wendroff explosion This test, also from [19], is computed on the domain $D = [0, 1.5] \times [0, 1.5]$ with reflecting inner boundaries and outflow outer boundaries using 256^2 elements. Inside the radius $r = \sqrt{(x^1)^2 + (x^2)^2} = 0.4$, we initially set $\mathbf{U} = (1.0, \mathbf{0.0}, 2.5)^T$, while $\mathbf{U} = (0.125, \mathbf{0.0}, 0.25)^T$ elsewhere. We did not apply any smoothing to the initial discontinuity.

Results for $t = 3.2$, obtained with $C_{\text{TCl}} = 0.1$, are displayed in Figure 7, which agree qualitatively with results in [19]. Initially, this test is similar to the Sod shock tube in cylindrical geometry: a shock and a contact discontinuity propagate in the positive radial direction, while a rarefaction wave propagates toward the origin. Later, the initial shock leaves the computational domain, while an instability (resembling Richtmeyer-Meshkov) develops in the contact discontinuity. A second shock emerges from the origin of the computational domain and runs through the contact discontinuity, which has developed a complex structure (see density contours in Figure 7). The second shock is located around $r = 1.2$ at $t = 3.2$. In the right panel

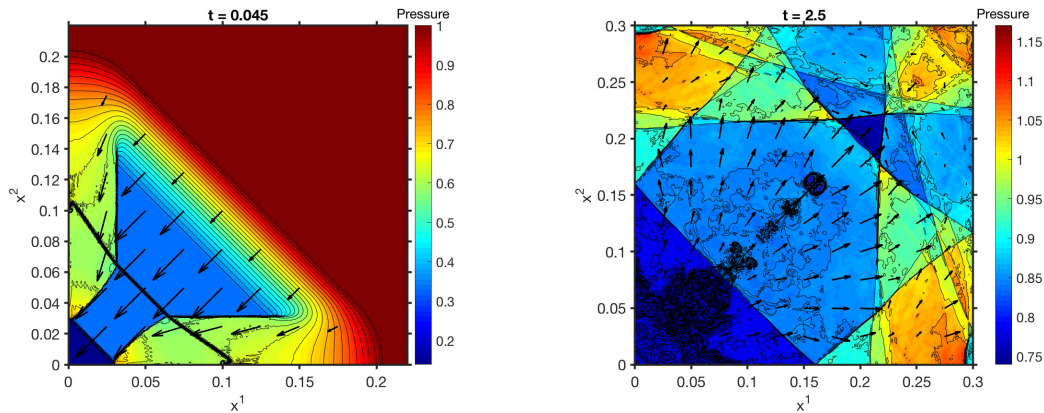


Figure 6. Plots for the Liska-Wendroff implosion problem at $t = 0.045$ (left) and $t = 2.5$ (right), intended to match Figures 4.10 and 4.11 in [19]. The color map shows the pressure distribution, black contours show the mass density, while arrows indicate the velocity.

we plot the distribution of elements flagged for limiting by the troubled-cell indicator at $t = 3.2$. Limiting is mainly focused on elements around the shock and the contact discontinuity.

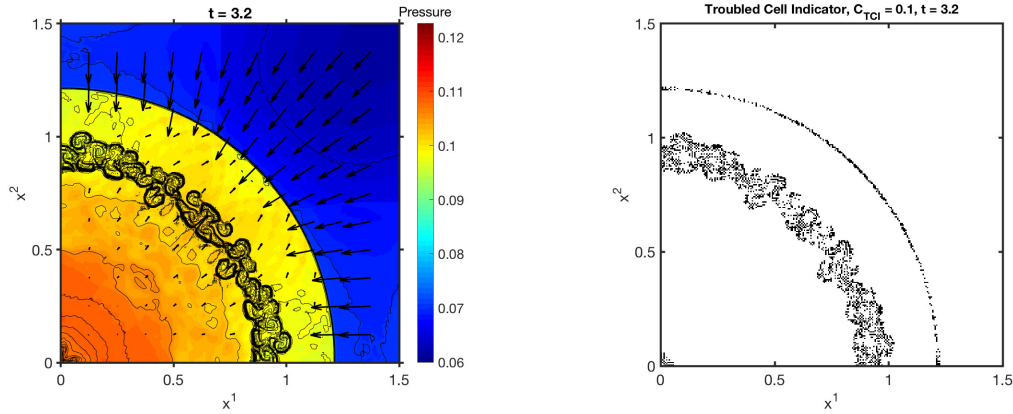


Figure 7. Plots for the Liska-Wendroff Explosion problem at $t = 3.2$. In the left panel, intended to match panels in Figure 4.12 in [19], the color map shows the pressure distribution, black contours show the mass density, while arrows indicate the velocity. In the right panel we plot the distribution of elements flagged for limiting $t = 3.2$.

3.1.8. Standing accretion shock (SAS) This test is of immediate relevance to simulation of CCSNe. The initial conditions, described in detail in [3], have been used by many to study the standing accretion shock instability (SASI) and CCSN explosion dynamics. We follow closely the description in [3] to initialize this test. We use spherical polar coordinates, let $D = [0.2, 2.0]$, and $\Gamma = 4/3$. The gravitational potential is given by the point-mass formula with $GM = 0.5$. A stationary shock is placed at a radius $R_{\text{sh}} = 1$. Ahead of the shock, the flow is essentially in free-fall towards the shock with a constant Mach number of 100. The mass accretion rate is held fixed at the outer boundary to $\dot{M} = 4\pi$. Below the shock, $r \in [0.2, 1]$, the settling solution is obtained by solving Bernoulli's equation. Matter flows through the inner boundary, and the density and pressure in boundary elements are set by extrapolating from D assuming $\rho \propto r^{-3}$ and $E \propto r^{-4}$, while the momentum components are held fixed to their initial values.

We use 256 elements and run the tests until $t = 100$ (> 5 dynamical times [3]). Results are shown in Figure 8. In the first test we show results from an unperturbed run to gauge the ability of the DG method to maintain the initial state. In the upper left panel we compare the initial state (dashed lines) with the solution at $t = 100$. Except for a slight shift ($\sim 1\%$) in the position of the shock, the initial and final states are indistinguishable on the plot. Also, there are no oscillations visible in the numerical results. In the upper right panel we show elements flagged for limiting in the rt -plane (we set $C_{\text{TCI}} = 0.03$ in this run), which illustrates how limiting is confined to the shock. In the second test we placed a shell with thickness 0.2 and density three times higher than the ambient density ahead of the shock to induce a strong radial perturbation (similar to [3]; see their Figure 4). The initial condition is stable against radial perturbations. In the lower panel in Figure 8 we plot the pressure deviation below the shock, defined as

$$\delta p = (p - \bar{p})/\bar{p}, \quad \text{where} \quad \bar{p} = p_0(r_0/r)^4, \quad (16)$$

and p_0 and r_0 are the pressure and radius at the inner boundary. As the shell falls through the shock, the strong perturbation results in an interesting pattern of waves propagating inside the shocked cavity, and oscillations of the shock position. The configuration eventually settles down

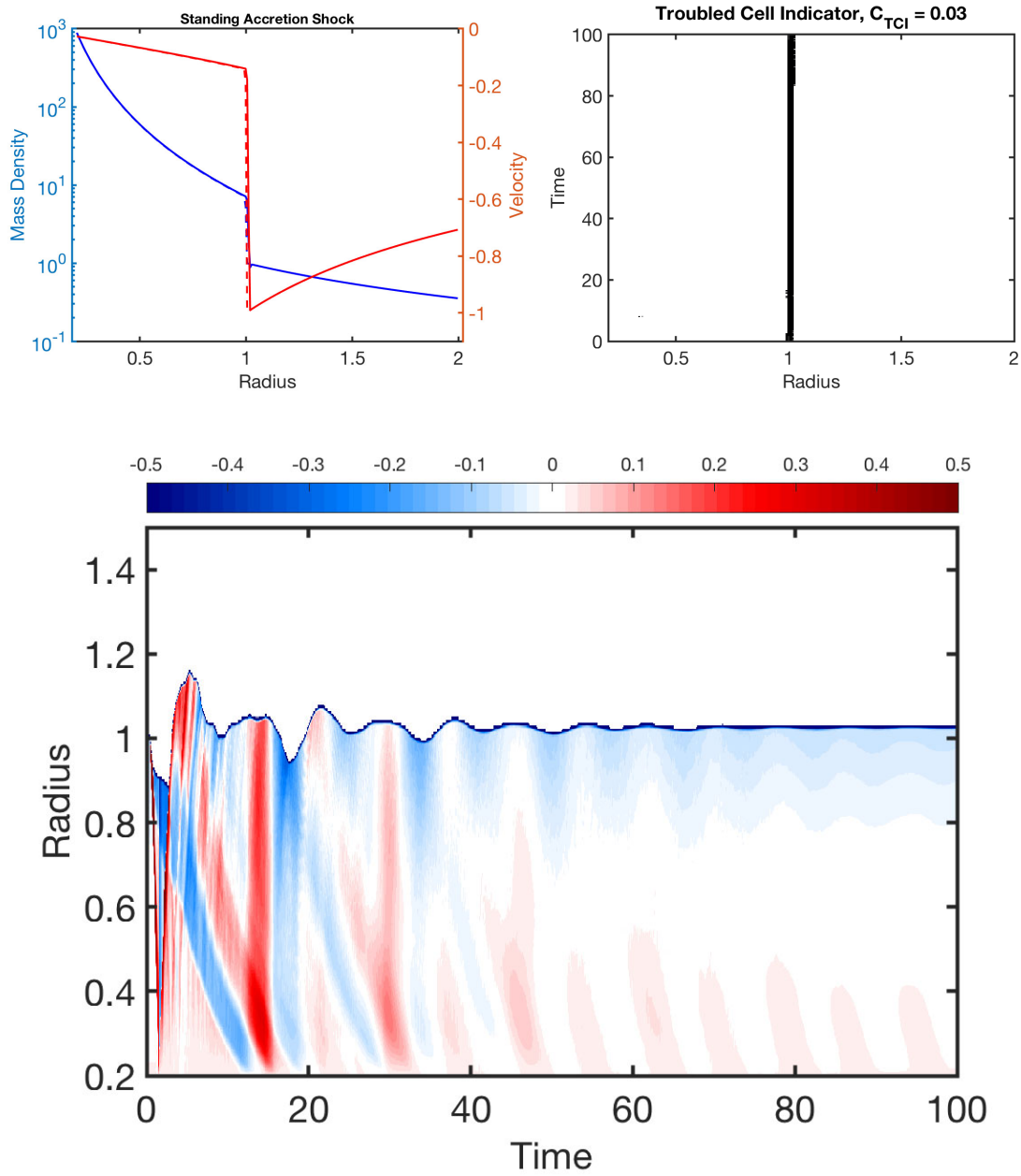


Figure 8. Results from the SAS test using 256 elements. In the upper left panel we plot the mass density (blue) and velocity (red) versus radius for the initial condition (dashed lines) and for $t = 100$ (solid lines) from an unperturbed run. In the upper right panel we show elements flagged for limiting in the rt -plane. In the bottom panel we show results from a perturbed model, where we plot the relative deviation of the pressure below the shock; cf. Eq. (16).

to a configuration that is close to the initial condition (although the position of the shock is slightly larger than that of the initial state.)

3.1.9. Yahil-Lattimer collapse The final test of the NR hydrodynamics in **thornado** involves self-gravity and is due to [41, 42]. It models the self-similar collapse of a polytropic star;

i.e., $p = \kappa \rho^\Gamma$, where κ is the polytropic constant. This test is also of immediate relevance to CCSN simulations. In [41, 42], self-similar solutions to the gravitational collapse problem were constructed for $6/5 \leq \Gamma < 4/3$. The solutions smoothly connect an homologously collapsing subsonic inner core (velocity proportional to radius) to a supersonically collapsing outer core in near free-fall ($u \propto r^{-1/2}$). With two dimensional parameters in the model (the gravitational

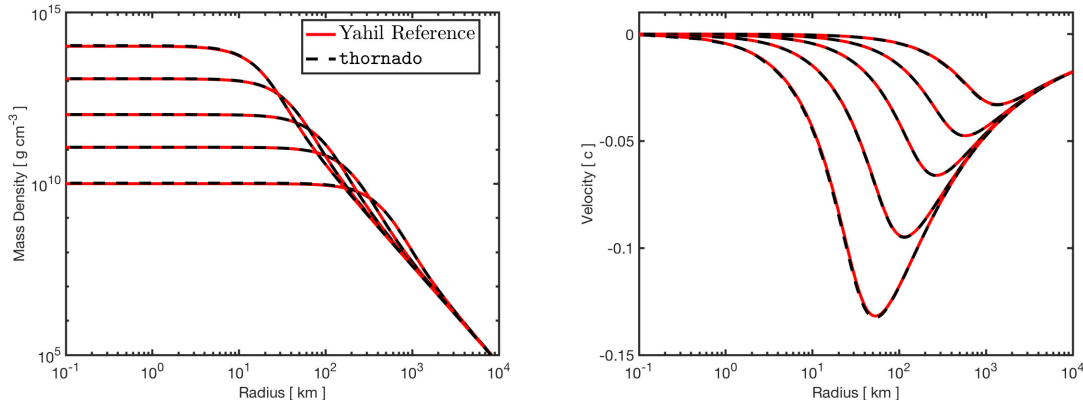


Figure 9. Results from the Yahil-Lattimer collapse test using 256 elements. Results obtained with `thornado` (dashed black) are compared with the reference solution from Yahil [42] (solid red). The mass density (left panel) and velocity (right panel) are plotted versus radius. We compare the solutions at select central densities during collapse, approximately $[10^{10}, 10^{11}, 10^{12}, 10^{13}, 10^{14}]$ g cm $^{-3}$, which correspond to $(-t) = [51.0, 15.0, 5.0, 1.5, 0.5]$ ms.

constant G and the polytropic constant κ), the dimensionless similarity variable is

$$X = \kappa^{-1/2} G^{(\Gamma-1)/2} r (-t)^{\Gamma-2}, \quad (17)$$

where the origin of time is the moment of infinite central density. All the hydrodynamic variables can be expressed as a function of X , and the time-dependent Euler equations can be recast as a system of ODEs (see [42] for details). To construct a reference to compare with our numerical results we have solved the ODEs given in [42] to obtain these self-similar solutions.

We show results for a model with $\Gamma = 1.30$. We use spherical polar coordinates, and let $D = [0, 1 \times 10^5]$ km, which is covered with 256 elements. The Newtonian gravitational potential is obtained by solving Poisson's equation using a third-order accurate continuous finite element method (e.g., [4]). We use a geometric grid to resolve the mass distribution as the star collapses and the central density increases from about 10^9 g cm $^{-3}$ to about 10^{14} g cm $^{-3}$. The size of the innermost element is set to 1 km while the size of the last element is about 3×10^3 km. We specify the polytropic constant by setting a reference pressure $p = 6 \times 10^{27}$ erg cm $^{-3}$ for $\rho = 7 \times 10^9$ g cm $^{-3}$ (reasonable values for a massive star in the pre-collapse stage). We also set the collapse time to $(-t) = 150$ ms. Results comparing the mass density and velocity of Yahil's reference solution to those obtained with `thornado` using $C_{\text{TCI}} = 0.03$ are plotted in Figure 9. The agreement of the mass density and velocity profiles is excellent throughout collapse.

3.2. Special relativistic (SR) hydrodynamics

Here we show results from solving the SR Euler equations in Cartesian coordinates with `thornado`. In this case the state, flux, and source vectors in Eq. (1) are

$$\mathbf{U} = (\rho W, \rho h W^2 u_j, \tau)^T, \quad \mathbf{F}^i = (\rho W u^i, \Pi^i_j, \rho(hW - 1)W u^i)^T, \quad \text{and} \quad \mathbf{S} = 0, \quad (18)$$

where W is the Lorentz factor, $h = 1 + (e + p)/\rho$ is the specific enthalpy, $\tau = \rho W(hW - 1) - p$, and $\Pi_j^i = \rho h W^2 u^i u_j + p \delta_j^i$. To recover primitive from conserved variables we have implemented the procedure for analytic equations of state detailed in Appendix D in [27]. We solve two Riemann problems and compare the numerical results with exact solutions obtained with the solver from [20]. Let the primitive state vector be $\mathbf{V} = (\rho, \mathbf{u}, p)^T$. In the first problem (Riemann problem 1), taken from [22], we use $\Gamma = 4/3$ and

$$\mathbf{V}_L = (1.0, 0.9, 0.0, 0.0, 1.0)^T \quad \text{and} \quad \mathbf{V}_R = (1.0, 0.0, 0.0, 0.0, 10.0)^T,$$

while in the second problem (Riemann problem 2), also from [22], we use $\Gamma = 5/3$, and let

$$\mathbf{V}_L = (1.0, \mathbf{0.0}, 10^3)^T \quad \text{and} \quad \mathbf{V}_R = (1.0, \mathbf{0.0}, 10^{-2})^T.$$

In both problems, the computational domain is $D = [0, 1]$, the discontinuity is initially located at $x^1 = 0.5$, and the solutions are integrated to $t = 0.4$. Results are plotted in Figure 10.

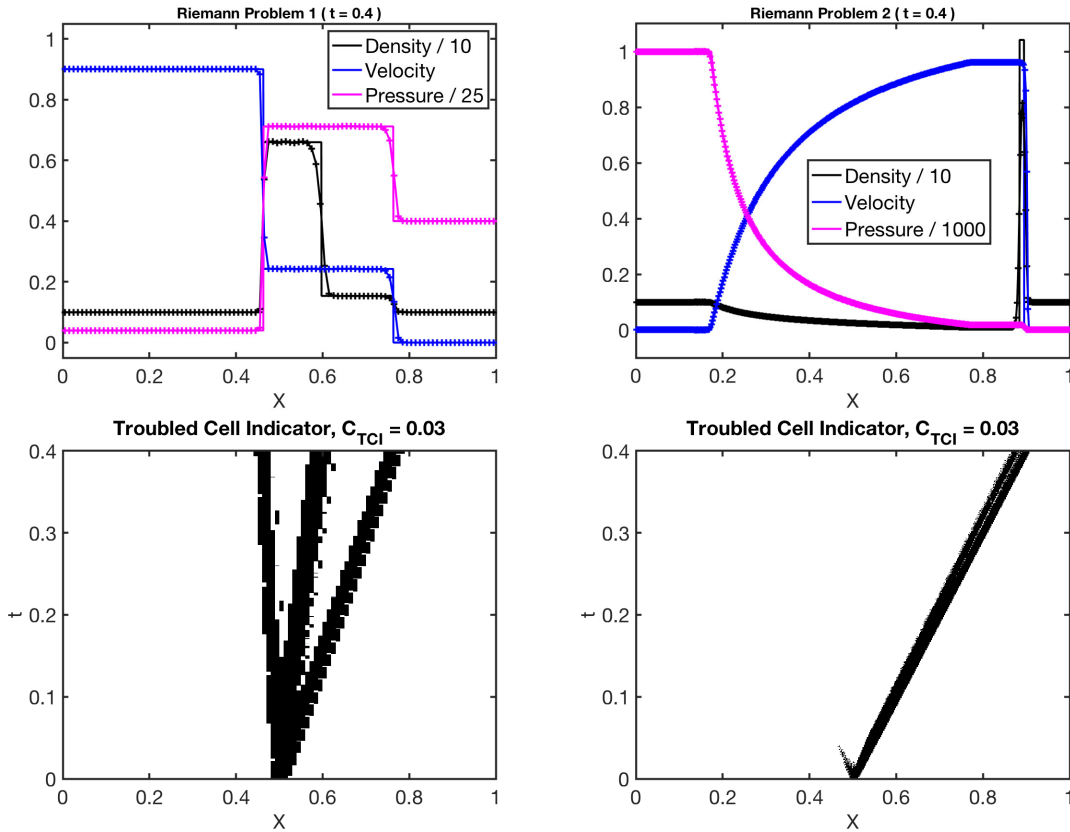


Figure 10. Results from solving the SR Euler equations with *thornado*. Riemann problem 1 (left panels) was solved using 100 elements, while Riemann problem 2 (right panels) was solved using 400 elements. The numerical and exact solutions are plotted with plusses and solid lines, respectively. In the lower panels we plot xt -plane locations of elements flagged for limiting

Again, the main features of the exact solution are captured with the DG method implemented in *thornado*. For Riemann problem 1, we observe some oscillations in the density profile between the leftmost shock and the contact discontinuity located around $x^1 = 0.6$. For Riemann problem 2, the maximum density in the thin density shell between the shock and the contact

discontinuity is significantly lower for the numerical solution than the exact solution. This is due to a combination of limited spatial resolution and the action of the slope limiter. We used $C_{\text{TCI}} = 0.03$ in both tests in this section, and we plot locations in the xt -plane of elements flagged for limiting in Figure 10. The troubled-cell indicator tracks the discontinuities well for both Riemann problems. We have found the agreement with the exact solution to improve for larger values of C_{TCI} , but at the expense of somewhat more oscillatory results.

4. Summary and outlook

We have presented preliminary algorithm details and numerical results for solvers of the non-relativistic and special relativistic Euler equations of gas dynamics as implemented in the toolkit for **high-order** neutrino-radiation hydrodynamics (**thornado**). The spatial discretization is based on the DG method and the ODEs resulting from this discretization are integrated in time using SSP-RK methods. We employ a spectral-type nodal collocation DG approximation based on Legendre-Gauss points for interpolation and numerical quadrature evaluation [2]. This choice simplifies expressions for the semi-discretized equations, especially when applied to problems involving curvilinear coordinates encoded in a metric (e.g., in numerical relativity). Results from a suite of tests in one and two spatial dimensions — including problems with strong shocks — demonstrate reliable performance of the implementation. The accuracy of **thornado** on three shock tests (Sod shock tube [34], Sedov-Taylor blast wave [29], and Shu-Osher shock tube [31]) is qualitatively similar to that of the well-established CCSN simulation code CHIMERA [5], which is based on the PPM finite volume method [8]. Two tests in spherical symmetry (standing accretion shock [3] and Yahil-Lattimer collapse [41, 42]) demonstrate the DG method’s ability to handle conditions relevant to CCSN simulations. We will present a more in-depth analysis of the performance of **thornado** on these (and related) problems in a future study.

The performance of the DG algorithm is sensitive to limiting of the polynomial representation. The combination of a TVD-type limiter (e.g., [6]) and the troubled-cell indicator of [12] seems to give satisfactory results for a range of problems. However, in our experience, the optimal value for the indicator threshold C_{TCI} seems to vary with the specific problem, and further investigation is needed to determine if there exists an optimal value suitable for CCSN simulations. Along these lines of investigation, it would also be interesting to explore and compare the use of the *a posteriori* subcell limiting approach in [9, 11] with our current approach.

Ongoing and planned near-future work within **thornado** include coupling to solvers for two-moment neutrino transport, extensions to accommodate nuclear equations of state, general relativity within the conformal flatness approximation, and deployment within an adaptive mesh refinement framework. We hope to report on progress in these directions in the near future.

Acknowledgments

Eirik Endeve, Anthony Mezzacappa, Nick Roberts, and Samuel J. Dunham acknowledge support from the NSF Gravitational Physics Program (NSF-GP 1505933 and 1806692).

References

- [1] Ascher U M, Ruuth S J and Spiteri R J 1997 *Applied Numerical Mathematics* **23** 151
- [2] Bassi F, Franchina N, Ghidoni A and Rebay S 2013 *Int. J. Numer. Meth. Fluids* **71** 1322
- [3] Blondin J M, Mezzacappa A and DeMarino C 2003 *ApJ* **584** 980
- [4] Brenner S C and Scott L R 2008 *The Mathematical Theory of Finite Element Methods* Springer
- [5] Bruenn S W *et al Preprint* arXiv:1809.05608
- [6] Cockburn B and Shu C-W 1998 *JCP* **141** 199
- [7] Cockburn B and Shu C-W 2001 *Journal of Scientific Computing* **16** 173
- [8] Colella P and Woodward P 1984 *JCP* **54** 174
- [9] Dumbser M, Zanotti O, Loubère R and Diot S 2014 *JCP* **278** 47
- [10] Dumbser M, Guercilena F, Köppel S, Rezzolla L and Zanotti O 2018 *Phys. Rev. D* **97** 084053

- [11] Fambri F, Dumbser M, Köppel S, Rezzolla L and Zanotti O 2018 *MNRAS* **477** 4543
- [12] Fu G and Shu C-W 2017 *JCP* **347** 305
- [13] Harten A, Lax P D and Van Leer B 1983 *SIAM Review* **25** 35
- [14] Hesthaven J S and Warburton T 2008 *Nodal discontinuous Galerkin methods: Algorithms, analysis and applications* Springer
- [15] Huerta A, Casoni E and Peraire J 2012 *Int. J. Numer. Meth. Fluids* **69** 1614
- [16] Kamm J R and Timmes F X 2007 *LANL Report, No. LA-UR-07-2849*
- [17] Klöckner A, Warburton T, Bridge J and Hesthaven J S 2009 *JCP* **228** 7863
- [18] Landau L D and Lifshitz E M 1979 *Fluid Mechanics* Pergamon
- [19] Liska R and Wendroff B 2003 *SIAM J. Sci. Comput.* **25** 3
- [20] Martí J M and Müller E 2003 *Living Rev. Relativ.* **6** 7
- [21] McNally C, Lyra W and Passy J-C 2012 *ApJS* **201** 18
- [22] Mignone A and Bodo G 2005 *MNRAS* **364** 126
- [23] Müller B 2016 *PASA* **33** 1
- [24] Qin T, Shu C-W and Yang Y 2016 *JCP* **315** 323
- [25] Radice D and Rezzolla L 2011 *Phys. Rev. D* **84** 024010
- [26] Remacle J-F, Flaherty J E and Shephard M S 2003 *SIAM Review* **45** 53
- [27] Rezzolla L and Zanotti O 2013 *Relativistic Hydrodynamics* Oxford
- [28] Schaal K, Bauer A, Chandrashekar P, Rüdiger P, Klingenberg C and Springel V 2015 *MNRAS* **453** 4278
- [29] Sedov L I 1959 *Similarity and Dimensional Methods in Mechanics* Academic Press
- [30] Shu C-W and Osher S 1988 *JCP* **77** 439
- [31] Shu C-W and Osher S 1989 *JCP* **83** 32
- [32] Shu C-W 1997 *NASA/CR-97-206253 ICASE Report No. 97-65*
- [33] Shu C-W 2016 *JCP* **316** 598
- [34] Sod G A 1978 *JCP* **27** 1
- [35] Sonntag M and Munz C-D 2017 *Journal of Scientific Computing* **70** 1262
- [36] Teukolsky S A 2016 *JCP* **312** 333
- [37] Toro E F, Spruce M and Speares W 1994 *Shock Waves* **4** 25
- [38] Toro E F 1999 *NUMERICA: A Library of Source Codes for Teaching, Research and Applications*
- [39] Wilson J R, Mathews G J and Marronetti P 1996 *Phys. Rev. D* **54** 1317
- [40] Wu K and Tang H 2015 *JCP* **298** 539
- [41] Yahil A and Lattimer J M 1982 *Supernovae: A Survey of Current Research* Reidel
- [42] Yahil A 1983 *ApJ* **265** 1047
- [43] Zanotti O, Fambri F and Dumbser M 2015 *MNRAS* **452** 3010
- [44] Zhang X and Shu C-W 2010 *JCP* **229** 8918