Towards Positivity Preservation for Monolithic Two-way Solid-Fluid Coupling

Saket Patkar*, Mridul Aanjaneya**, Wenlong Lu*, Michael Lentine*, Ronald Fedkiw*

Stanford University, 353 Serra Mall, Gates Computer Science Room 207, Stanford, CA 94305

University of Wisconsin - Madison, 1210 West Dayton St, Computer Science Room 1349, Madison, WI 53706

Abstract

We consider complex scenarios involving two-way coupled interactions between compressible fluids and solid bodies under extreme conditions where monolithic, as opposed to partitioned, schemes are preferred for maintaining stability. When considering such problems, spurious numerical cavitation can be quite common and have deleterious consequences on the flow field stability, accuracy, etc. Thus, it is desirable to devise numerical methods that maintain the positivity of important physical quantities such as density, internal energy and pressure. We begin by showing that for an arbitrary flux function, one can put conditions on the time step in order to preserve positivity by solving a linear equation for density fluxes and a quadratic equation for energy fluxes. Our formulation is independent of the underlying equation of state. After deriving the method for forward Euler time integration, we further extend it to higher order accurate Runge-Kutta methods. Although the scheme works well in general, there are some cases where no lower bound on the size of the allowable time step exists. Thus, to prevent the size of the time step from becoming arbitrarily small, we introduce a conservative flux clamping scheme which is also positivity preserving. Exploiting the generality of our formulation, we then design a positivity preserving scheme for a semi-implicit approach to time integration that solves a symmetric positive definite linear system to determine the pressure associated with an equation of state. Finally, this modified semi-implicit approach is extended to monolithic two-way solid-fluid coupling problems for modeling fluid structure interactions such as those generated by blast waves impacting complex solid objects.

1. Introduction

We consider simulating fluid structure problems, especially those involving complex interactions, which require the use of robust monolithic approaches [34, 14, 33, 11, 13] in order to guarantee stability and accuracy. In particular, we utilize the semi-implicit approach for compressible flow introduced in [22] and extended to two-way solid-fluid coupling in [14, 13]. When considering extreme scenarios such as blast waves interacting with complex objects, numerical schemes often fail when the density or internal energy becomes negative creating an imaginary value for the sound speed and a non-physical solution. To obtain physically admissible solutions, the density and the internal energy should remain positive. Numerical methods that guarantee positivity of these quantities at all times are called positivity preserving. In fact, many commonly used high order accurate schemes for solving systems of hyperbolic conservation laws [16, 37, 5, 25, 20, 6] are not positivity preserving.

A common ad hoc approach for maintaining positivity is to clamp the density, pressure, or internal energy if any of these quantities goes below a certain threshold. However, this destroys local conservation and can produce highly inaccurate results, see e.g. [44]. Moreover, clamping a variable in one time step can lead to negativity of other variables in subsequent time steps. Although the density can be clamped in a

*{patkar,wenlongl,mlentine,fedkiw}@cs.stanford.edu, Stanford University

**mridul.aanjaneya@wisc.edu, University of Wisconsin - Madison

straightforward manner, it is not clear how to clamp the internal energy, i.e., should one clamp the total energy, the momentum, or both, etc. We experimented with several different options on a large number of difficult examples, but were unable to find a consistent approach that works well across all examples or was successfully advocated in the literature. The main difficulty we encountered was that clamping degrades the solution accuracy, and the degradation cascades destroying the entire solution. The most important negative consequence of clamping is that the resulting solution may actually seem to work just fine providing a plausible albeit completely wrong and misleading result. Consider, for example, a rigid block moving in a constant density gas producing a shock wave in front of it and a wake behind it. This example is considered and detailed in Sections 6.2 and 7.3. Figure 1(a) shows the results utilizing minimal clamping with lower bounds $\rho_{\texttt{min}}$ and $e_{\texttt{min}}$ in the equation of state and other calculations that require positive values, while leaving the values otherwise unchanged in order to maintain conservation. Although clamping is used on only a few grid cells for only a few time steps, the result is still quite inaccurate compared to the ground truth shown in Figures 9 and 19.

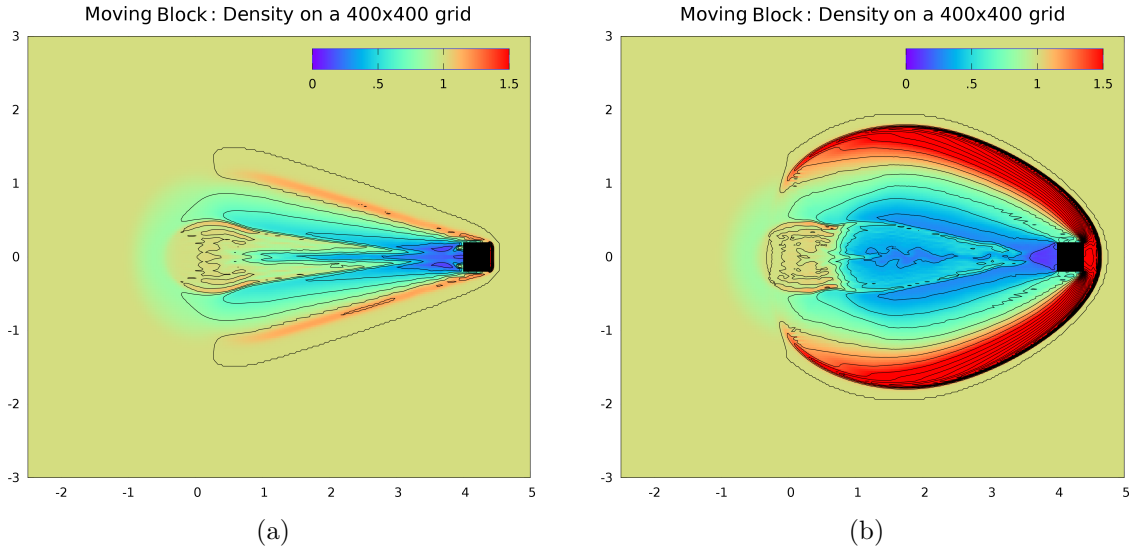


Figure 1: Density contours for the moving block example from Section 7.3 (a) clamping both the density and the internal energy, and (b) using our method which adaptively clamps the time steps and the fluxes, and closely matches the ground truth.

For one-dimensional scalar conservation laws, the problem of designing robust high order accurate positivity preserving schemes is well-posed, since the entropy solution satisfies the total variation diminishing (TVD) property [35, 39]. The TVD property is more difficult to enforce for multi-dimensional scalar conservation laws; however, the entropy solution satisfies a strict maximum principle and high order accurate schemes which have this property have been designed [40, 42]. The compressible Euler equations are a system of hyperbolic conservation laws for which the entropy solution in general satisfies neither the TVD property nor the maximum principle. First order accurate schemes such as the Godunov exact Riemann solver [8], the Steger-Warming scheme [12], Einfeld's modification [7] to the Harten-Lax-van Leer scheme [17], the Lax-Friedrichs scheme [30], and the Boltzmann type scheme [29] have been shown to be positivity preserving under a CFL-like condition. However, these schemes lose the positivity preserving property when extended to second order accuracy using flux limiters or MUSCL slope limiters [3]. Thus, various researchers have investigated the design of high order accurate schemes which maintain the positivity preserving property [30, 1, 2]. These methods are computationally quite expensive, and the CFL-like condition can be difficult to enforce in practice. Positivity preserving flux-limiting schemes have been proposed using the rule of positive coefficients [26, 27], which requires the coefficients of the discrete equations to have positive eigenvalues and eigenvectors identical to those of the Jacobian matrix for the corresponding flux vectors.

However, this method is limited to the gamma law gas equation of state. A positivity preserving limiter for high order accurate discontinuous Galerkin (DG) schemes on rectangular meshes was proposed in [41]. This scheme was later extended to triangular meshes [45], to the Euler equations with source terms and a general equation of state [43], to high order ENO and WENO schemes [44], and to Lagrangian schemes for multi-material compressible flow [4].

Instead of addressing positivity one scheme, order of accuracy, or equation of state at a time, our goal is to derive some general techniques that can be applied to any scheme and any flux function, similar in spirit to [19]. This is motivated in part by the increasing complexity of numerical methods as the community focuses on increasingly difficult problems including all speed flows and two-way solid-fluid coupling. We show that for an arbitrary flux function one can derive conditions on the time step that preserve positivity by solving a linear equation for density fluxes and a quadratic equation for internal energy. Although we only consider the gamma law gas equation of state, there are no assumptions within our formulation that limit its applicability to other equations of state. The scheme is first derived for forward Euler time integration and then extended to higher order accurate TVD Runge-Kutta methods [37, 38]. This is achieved by treating the intermediate Runge-Kutta updates as independent forward Euler steps so that the linear and quadratic equations can still be used. Though the approach works well in general, the lack of a lower bound on the size of the allowable time step can prove problematic in certain scenarios. Thus, we introduce a conservative flux clamping scheme which is applied locally in space and time in order to preserve positivity when one wishes to avoid any further refinement of the size of the time step. While this scheme does degrade the overall spatial accuracy of the solution (which can be avoided by combining the clamped flux with a first order accurate flux, see [19] for details), it readily generalizes to our main goal of applying this methodology to increasingly complex problems such as those arising in two-way coupled fluid structure interactions. Since these methods are most robust when monolithic, we also extend our method to a compressible flow solver that implicitly solves for a pressure consistent with the underlying equation of state. Subsequently, we demonstrate the efficacy of our approach on quite difficult problems involving two-way solid-fluid coupling. We modify the method of [33] to treat the mass in the solid-fluid region correctly, and also extend it to handle objects that are under-resolved on the background grid using ideas from [28].

2. Compressible Euler equations

The compressible Euler equations in multiple spatial dimensions are defined as follows

$$\mathbf{U}_t + \nabla \cdot \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho \\ \rho\vec{u} \\ E \end{pmatrix}_t + \nabla \cdot \begin{pmatrix} \rho\vec{u} \\ \rho\vec{u} \otimes \vec{u} + p \\ (E+p)\vec{u} \end{pmatrix} = 0 \tag{1}$$

where the conserved variables are the density ρ, momentum $\rho\vec{u}$, and the total energy E. The total energy can be written as

$$E = \rho e + \frac{(\rho\vec{u}) \cdot (\rho\vec{u})}{2\rho} \tag{2}$$

where e is the internal energy per unit mass. A state $\mathbf{U} = (\rho, \rho\vec{u}, E)$ satisfies positivity if both ρ and e are positive, whilst $\vec{u}$ can be of any sign. For an arbitrary equation of state, the pressure p typically depends on ρ and e but not $\vec{u}$. Let p_ρ and p_e denote the partial derivatives of p with respect to ρ and e at constant $\vec{u}$, then the sound speed c emanates from the eigenvalues of the Jacobian matrix,

$$c = \sqrt{p_\rho + p p_e / \rho^2}. \tag{3}$$

See e.g. [10] and the references therein.

3. Adaptive time step restriction

Consider a forward Euler temporal update in one spatial dimension for cell i,

$$\begin{aligned}
\rho_i^{n+1} &= \rho_i^n - \Delta t \frac{\mathcal{D}_{i+1/2}^n - \mathcal{D}_{i-1/2}^n}{\Delta x} = \rho_i^n - \Delta t \mathbf{D}_i^n \\
(\rho\vec{u})_i^{n+1} &= (\rho\vec{u})_i^n - \Delta t \frac{\vec{\mathcal{M}}_{i+1/2}^n - \vec{\mathcal{M}}_{i-1/2}^n}{\Delta x} = (\rho\vec{u})_i^n - \Delta t \vec{\mathbf{M}}_i^n \\
E_i^{n+1} &= E_i^n - \Delta t \frac{\mathcal{E}_{i+1/2}^n - \mathcal{E}_{i-1/2}^n}{\Delta x} = E_i^n - \Delta t \mathbf{E}_i^n.
\end{aligned} \tag{4}$$

where $\mathcal{D}_{i+1/2}^n, \vec{\mathcal{M}}_{i+1/2}^n$, and $\mathcal{E}_{i+1/2}^n$ are the density, momentum, and energy fluxes at face $i+1/2$ at time t^n, and $\mathbf{D}_i^n$, $\vec{\mathbf{M}}_i^n$, and $\mathbf{E}_i^n$ are the density, momentum, and energy flux divided differences at grid cell i. We assume Δt_{CFL} is given by the regular CFL time step restriction, or is chosen based on accuracy concerns. In this section we further limit Δt below Δt_{CFL}, if needed, to maintain positivity.

If $\mathbf{D}_i^n$ is positive, the density is decreasing and for a large enough time step ρ_i^{n+1} will become negative. It is straightforward to choose a time step that maintains positivity of ρ_i^{n+1}. For the sake of accuracy and robustness, we limit the change in density during a time step so that it only shrinks to a fraction ε_ρ of its initial value, i.e.,

$$\rho_i^{n+1} = \rho_i^n - \Delta t \mathbf{D}_i^n \geq \varepsilon_\rho \rho_i^n$$

This equation can be rearranged to obtain a time step restriction,

$$\Delta t_\rho = \min_{i \in \Omega_\rho} \Delta t_{\rho_i} = \min_{i \in \Omega_\rho} \frac{(1-\varepsilon_\rho)\rho_i^n}{\mathbf{D}_i^n} \tag{5}$$

where the minimum is taken over all grid cells for which $\mathbf{D}_i^n$ is greater than zero. If $\mathbf{D}_i^n$ is negative or zero for all cells i or $\Delta t_\rho \geq \Delta t_{\text{CFL}}$, then we use Δt_{CFL} as usual.

Next, consider the internal energy e_i^{n+1} which we require to remain positive and not shrink beyond a constant factor ε_e of its initial value in any given time step. This gives rise to the following inequality

$$e_i^{n+1} = \frac{E_i^{n+1}}{\rho_i^{n+1}} - \frac{||(\rho\vec{u})_i^{n+1}||^2}{2(\rho_i^{n+1})^2} = \frac{E_i^n - \Delta t \mathbf{E}_i^n}{\rho_i^n - \Delta t \mathbf{D}_i^n} - \frac{||(\rho\vec{u})_i^n - \Delta t \vec{\mathbf{M}}_i^n||^2}{2(\rho_i^n - \Delta t \mathbf{D}_i^n)^2} \geq \varepsilon_e e_i^n \tag{6}$$

Multiplying through by $2(\rho_i^n - \Delta t \mathbf{D}_i^n)^2$ gives the following quadratic inequality

$$2(\rho_i^n - \Delta t \mathbf{D}_i^n)(E_i^n - \Delta t \mathbf{E}_i^n) - ||(\rho\vec{u})_i^n - \Delta t \vec{\mathbf{M}}_i^n||^2 \geq 2\varepsilon_e e_i^n (\rho_i^n - \Delta t \mathbf{D}_i^n)^2$$

Expanding out the terms and gathering the coefficients for Δt^2 and Δt gives an inequality of the following form

$$A\Delta t^2 - 2B\Delta t + C \geq 0 \tag{7}$$

where,

$$A = 2\mathbf{D}_i^n \mathbf{E}_i^n - ||\vec{\mathbf{M}}_i^n||^2 - 2\varepsilon_e e_i^n (\mathbf{D}_i^n)^2 \tag{8}$$

$$B = \mathbf{D}_i^n E_i^n + \rho_i^n \mathbf{E}_i^n - (\rho\vec{u})_i^n \cdot \vec{\mathbf{M}}_i^n - 2\varepsilon_e e_i^n \rho_i^n \mathbf{D}_i^n \tag{9}$$

$$C = 2\rho_i^n (E_i^n - \varepsilon_e e_i^n \rho_i^n) - ||(\rho\vec{u})_i^n||^2 \tag{10}$$

Substituting equation (2) into equation (10) gives a simplified expression for C as follows

$$C = 2\rho_i^n \left(\rho_i^n e_i^n + \frac{||(\rho\vec{u})_i^n||^2}{2\rho_i^n} - \varepsilon_e e_i^n \rho_i^n \right) - ||(\rho\vec{u})_i^n||^2 = 2(1-\varepsilon_e) e_i^n (\rho_i^n)^2 \tag{11}$$

Since ρ_i^n and e_i^n are positive and $\varepsilon_e < 1$, C is always positive and the inequality (7) is always true when $\Delta t = 0$. Let $\Delta t_1, \Delta t_2$ be the two roots when equation (7) is written with a strict equality[1]. If the roots are imaginary, inequality (7) always holds. Otherwise, assuming that $\Delta t_1 < \Delta t_2$, inequality (7) can be written as

$$A(\Delta t - \Delta t_1)(\Delta t - \Delta t_2) \geq 0 \tag{12}$$

If $A > 0$, then the quadratic is concave up and inequality (12) is true when $\Delta t \in (-\infty, \Delta t_1] \cup [\Delta t_2, \infty)$. Since $\Delta t_1 \Delta t_2 = C/A$, which is positive in this case, both roots are of the same sign. If $B < 0$ then both roots are negative, and no additional time step restriction is required. Otherwise if $B > 0$, then both roots are positive and we set $\Delta t_{e_i} = \Delta t_1 = (B - \sqrt{B^2 - AC})/A$, which is the smaller of the two roots. If $A < 0$, then the quadratic is concave down and inequality (12) holds when $\Delta t \in [\Delta t_1, \Delta t_2]$. Since $\Delta t_1 \Delta t_2 = C/A < 0$ in this case, the roots differ in sign and we set $\Delta t_{e_i} = (B - \sqrt{B^2 - AC})/A$ which is the positive root independent of the sign of B. Notice that we always choose the root $(B - \sqrt{B^2 - AC})/A$ for all cases that have a time step restriction. Finally, we set $\Delta t_e = \min_{i \in \Omega_e} \Delta t_{e_i}$ similar to equation (5), where Ω_e is the set of cells where a minimum time step Δt_{e_i} was defined.

It is typically important to consider the limiting cases when the coefficients approach zero, since they may lead to numerical inaccuracies. While C can approach zero only from the positive side, A and B can approach zero from either side. We consider all possible cases where C is large in Table 1, deferring the limit as C approaches 0 until Section 5. Here '$\gg 0$' indicates a positive coefficient that is reasonably bounded away from zero, '$\ll 0$' indicates a negative coefficient that is reasonably bounded away from zero, and ε denotes a small positive number approaching zero. Note that when A and B are small[2], their sign can be uncertain due to numerical inaccuracies. Thus, we design a strategy that is valid irrespective of their sign. To aid in the understanding of the limiting values in Table 1, Figure 2 shows plots for the root $(B - \sqrt{B^2 - AC})/A$ as a function of A and B.

First, consider the cases where A approaches 0. **Case 1** corresponds to the scenario where $B \gg 0$. The root approaches $C/2B$ either from above or below as shown in Figure 2(a). Hence we use $C/2B - \delta$ as a robust root, where δ is a small number which can be chosen iteratively if necessary. On the other hand, when $B \ll 0$ as in **Case 2** the inequality is always true as shown in Figure 2(b). Cases 3 and 4 correspond to B approaching zero while A is large in magnitude. The root in **Case 3** is imaginary as shown in Figure 2(c). The root in **Case 4** approaches $\sqrt{-C/A}$ either from above or below as shown in Figure 2(d). Hence, similar to Case 1, we use $\sqrt{-C/A} - \delta$ as a robust root. Finally in **Case 5**, when A and B are both small, C dominates making the inequality always true.

We choose the overall time step as

$$\Delta t = \min\{\Delta t_\rho, \Delta t_e, \Delta t_{\text{CFL}}\} \tag{13}$$

noting that in certain scenarios Δt can become arbitrarily close to zero. This is addressed via the flux-limiting techniques presented in Section 5.

[1] In practice, we found that using the quadratic formula

$$\Delta t = \frac{B \pm \sqrt{B^2 - AC}}{A}$$

to compute the roots is prone to numerical errors when B is close in magnitude to $\sqrt{B^2 - AC}$. Therefore, we use the common approach of de-rationalizing the quadratic in order to compute the root which would potentially have catastrophic cancellation (see for example [18]).

[2] One could use a constant threshold below which a number can be deemed as 'small' if everything occurs on the same scale. Otherwise, a number that is 'small' can become significant just by scaling all the other numbers by a constant. Thus, we define the concept of a number being 'small' in a more robust fashion as follows: we look at the magnitude of all terms on the right hand side of equations (8), (9) and (10) and compare them with the magnitude of the result for A, B and C on the left hand side. If the maximum magnitude on the right hand side is more than 12 orders of magnitude greater than the magnitude of the left hand side, then we have less than three digits of accuracy on the result and we deem the left hand side coefficients (i.e. A, B, and C) small and inaccurate with respect to double precision (which supports approximately 15 digits of accuracy)

Case	A	B	Root	Robust Δt_{e_i}
1	$\pm\varepsilon$	$\gg 0$	$C/2B + O(A)$	$C/2B - \delta$
2	$\pm\varepsilon$	$\ll 0$	inequality always true	Δt_{CFL}
3	$\gg 0$	$\pm\varepsilon$	imaginary	Δt_{CFL}
4	$\ll 0$	$\pm\varepsilon$	$\sqrt{-C/A} + O(B)$	$\sqrt{-C/A} - \delta$
5	$\pm\varepsilon$	$\pm\varepsilon$	inequality always true	Δt_{CFL}

Table 1: Limiting cases when the coefficients A and B approach zero while C is bounded away from zero. The Root column denotes the limiting value of the root $(B - \sqrt{B^2 - AC})/A$. The Robust Δt_{e_i} column denotes the numerically robust value that can be assigned to Δt_{e_i}. δ is a small positive number.

4. TVD Runge-Kutta

We consider both second and third order accurate TVD Runge-Kutta (RK) temporal evolution [37, 38]. First, note that if states $\mathbf{U}_1 = (\rho_1, \vec{m}_1, E_1)$ and $\mathbf{U}_2 = (\rho_2, \vec{m}_2, E_2)$ are positivity preserving, where $\vec{m}_1 = (\rho\vec{u})_1$ and $\vec{m}_2 = (\rho\vec{u})_2$, then any linear combination $a\mathbf{U}_1 + b\mathbf{U}_2$ with $a, b \geq 0$ is also positivity preserving. The density $\rho = a\rho_1 + b\rho_2$ is trivially positive. The potential energy can be written as

$$\begin{aligned}
\rho e &= aE_1 + bE_2 - \frac{1}{2}\frac{||a\vec{m}_1 + b\vec{m}_2||^2}{a\rho_1 + b\rho_2} \\
&= a\left(\rho_1 e_1 + \frac{||\vec{m}_1||^2}{2\rho_1}\right) + b\left(\rho_2 e_2 + \frac{||\vec{m}_2||^2}{2\rho_2}\right) - \frac{1}{2}\frac{||a\vec{m}_1 + b\vec{m}_2||^2}{a\rho_1 + b\rho_2} \\
&= a\rho_1 e_1 + b\rho_2 e_2 + \frac{1}{2}\left(\frac{a||\vec{m}_1||^2}{\rho_1} + \frac{b||\vec{m}_2||^2}{\rho_2} - \frac{||a\vec{m}_1 + b\vec{m}_2||^2}{(a\rho_1 + b\rho_2)}\right)
\end{aligned}$$

where the first two terms are obviously positive, and the last term can be rewritten as

$$\frac{1}{2(a\rho_1 + b\rho_2)\rho_1\rho_2}((a\rho_2||\vec{m}_1||^2 + b\rho_1||\vec{m}_2||^2)(a\rho_1 + b\rho_2) - \rho_1\rho_2||a\vec{m}_1 + b\vec{m}_2||^2)$$

which can be shown to be equal to

$$\frac{ab}{2(a\rho_1 + b\rho_2)\rho_1\rho_2}||\rho_1\vec{m}_2 - \rho_2\vec{m}_1||^2$$

which is always non-negative. Alternatively, noting that the potential energy is a concave function of conserved variables, its convex combination will not affect positivity due to Jensen's inequality, similar in spirit to [41, 44].

Figure 3(a) illustrates standard TVD RK-2. The scheme uses two forward Euler steps to compute the intermediate state $\hat{\phi}^{n+2}$, before averaging ϕ^n and $\hat{\phi}^{n+2}$ in order to obtain ϕ^{n+1}, i.e.,

$$\phi^{n+1} = \frac{\phi^n + \hat{\phi}^{n+2}}{2} = \phi^n + \Delta t\left(\frac{\hat{\phi}^{n+2} - \phi^n}{2\Delta t}\right) \tag{14}$$

This update can equivalently be viewed as starting at the point ϕ^n and moving a distance Δt along the slope $(\hat{\phi}^{n+2} - \phi^n)/(2\Delta t)$. Obviously, if $\hat{\phi}^{n+2}$ is negative enough, then moving a distance Δt along the slope will also produce negative values for ϕ^{n+1}. Instead, we compute each of the two forward Euler time steps using the positivity preserving adaptive time step restriction given in Section 3 to obtain

$$\phi^{n+1} = \phi^n + \Delta t\left(\frac{\hat{\phi}^{n+2} - \phi^n}{\Delta t_1 + \Delta t_2}\right) \tag{15}$$

in place of equation (14). This can be rewritten as

$$\phi^{n+1} = \phi^n + \Delta t\left(\frac{\Delta t_1 L_1 + \Delta t_2 L_2}{\Delta t_1 + \Delta t_2}\right) \tag{16}$$

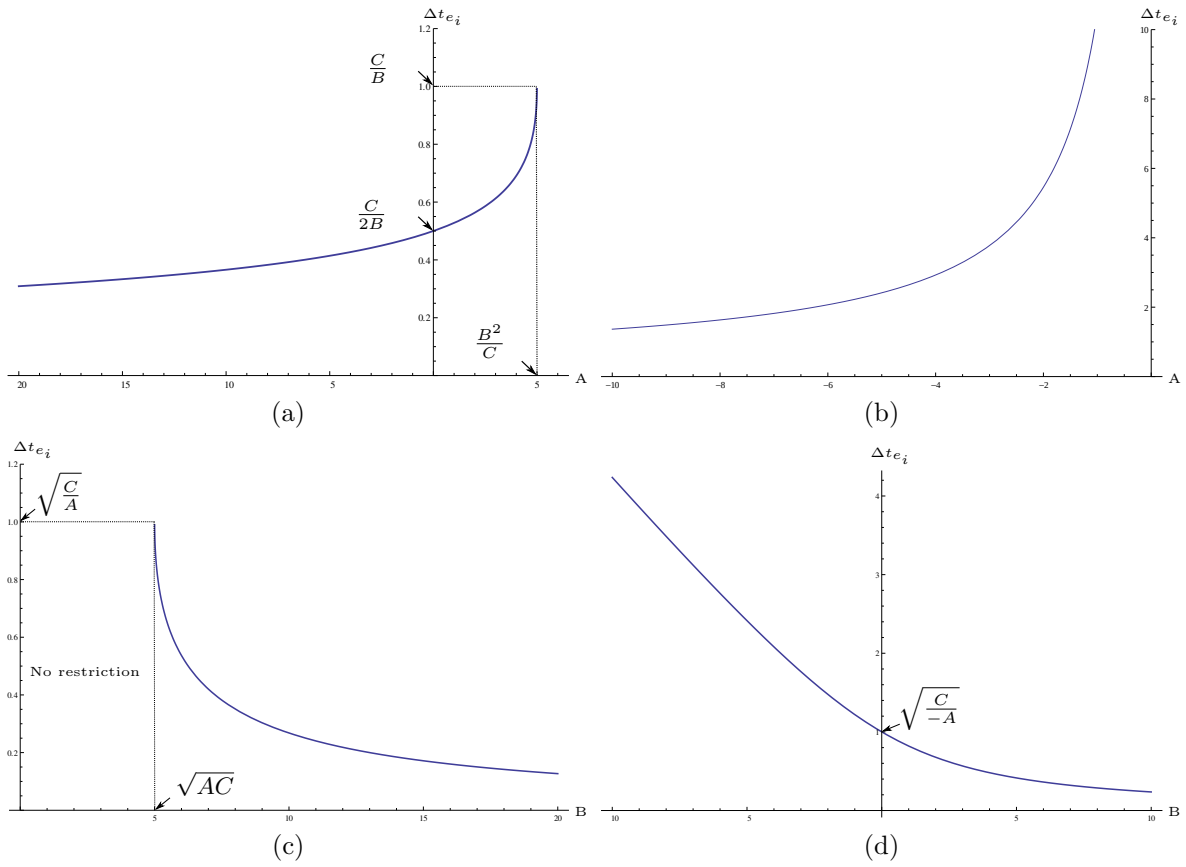


Figure 2: Representative plots for the maximum allowable time step size Δt_{e_i}. (a) Δt_{e_i} as a function of A for $B > 0$. $B = C = 5$ cross section is shown. Note that there is no restriction for $A > B^2/C$ (b) Δt_{e_i} as a function of A for $B < 0$. $B = -5, C = 5$ cross section is shown. Note that there is no restriction for $A > 0$. (c) Δt_{e_i} as a function of B for $A > 0$. $A = C = 5$ cross section is shown. Note that there is no restriction for $B < \sqrt{AC}$. (d) Δt_{e_i} as a function of B for $A < 0$. $A = -5, C = 5$ cross section is shown.

where L_1 and L_2 are the negative flux divided differences for the first and second Euler steps respectively. When compared to the conditions for TVD RK-2 from [37], we identify $\beta_{21} = \Delta t_2/(\Delta t_1 + \Delta t_2)$ and $\beta_{10} = \Delta t_1/\Delta t$, and use the fact that $2\beta_{21}\beta_{10} = 1$ to obtain $\Delta t = 2\Delta t_1 \Delta t_2/(\Delta t_1 + \Delta t_2)$ or

$$\phi^{n+1} = \phi^n + \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}\left(\frac{\hat{\phi}^{n+2} - \phi^n}{\Delta t_1 + \Delta t_2}\right) \tag{17}$$

in place of equation (15) as our final TVD RK-2 scheme. Since $2\Delta t_1 \Delta t_2 < (\Delta t_1 + \Delta t_2)^2$, the coefficients of ϕ^n and $\hat{\phi}^{n+2}$ are both positive, and so ϕ^{n+1} is positivity preserving.

As a test for the order of accuracy, we ran convergence analysis under temporal refinement on $y' = -y$ obtaining the expected results as shown in Table 2. We also provide a Taylor series analysis in Appendix I.

Next, consider standard TVD RK-3 which takes two forward Euler steps to compute $\hat{\phi}^{n+2}$ from ϕ^n, and then computes $\hat{\phi}^{n+1/2}$ as

$$\hat{\phi}^{n+1/2} = \frac{3}{4}\phi^n + \frac{1}{4}\hat{\phi}^{n+2} = \phi^n + \frac{\Delta t}{2}\left(\frac{\hat{\phi}^{n+2} - \phi^n}{2\Delta t}\right) \tag{18}$$

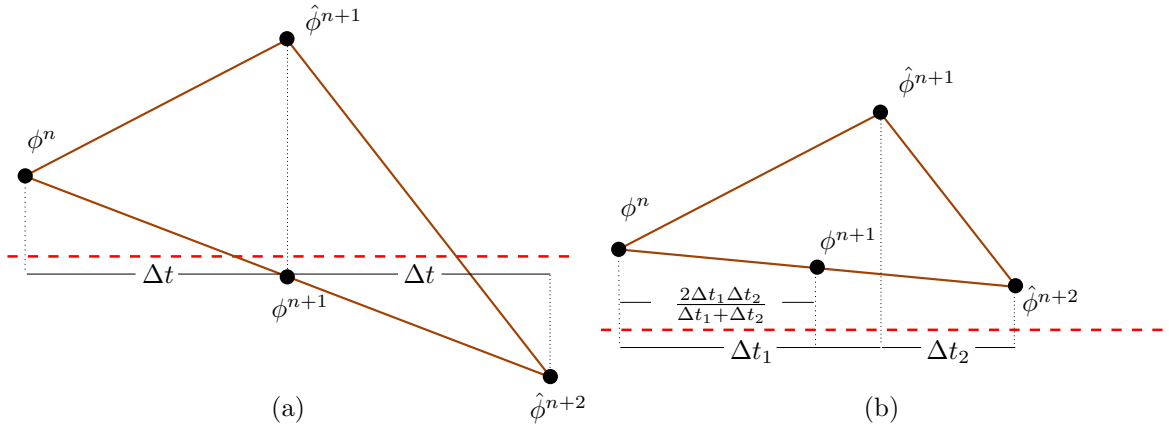


Figure 3: (a) Standard TVD RK-2 where states above the red line are positivity preserving. In the traditional scheme, $\hat{\phi}^{n+1}$ can be guaranteed to be positive using our adaptive time restriction. However, since the time step for the second RK step is the same as the first, $\hat{\phi}^{n+2}$ can be negative potentially making the state ϕ^{n+1} negative as well. (b) Our newly proposed TVD RK-2 where states above the red line are positivity preserving. This method computes a positivity preserving state ϕ^{n+1} by taking two different adaptive time steps Δt_1 and Δt_2, and subsequently moving along the slope by the distance $2\Delta t_1 \Delta t_2/(\Delta t_1 + \Delta t_2)$.

	Our RK-2 with fixed Δt_2		Our RK-2 with variable Δt_2		Standard RK-2	
Δt_1 $\times 10^{-2}$	error	convergence order	error	convergence order	error	convergence order
1	2.01503e-06	-	2.6709e-06	-	4.54522e-06	-
.5	5.02497e-07	2.00362	6.70571e-07	1.99386	1.13204e-06	2.00543
.25	1.25467e-07	2.00181	1.6724e-07	2.00347	2.82478e-07	2.00271
.125	3.13472e-08	2.0009	4.20045e-08	1.9933	7.05533e-08	2.00135
.0625	7.83435e-09	2.00045	1.0511e-08	1.99864	1.763e-08	2.00068
.03725	1.95828e-09	2.00022	2.61701e-09	2.00591	4.40648e-09	2.00034
.018625	4.89532e-10	2.00011	6.5513e-10	1.99807	1.10149e-09	2.00017

Table 2: Temporal convergence orders for the proposed version of TVD RK-2 for $y' = -y$. Here we choose $\Delta t_2 = .5\Delta t_1$ and show the errors and convergence orders in the second and third columns. The next two columns show results when choosing $\Delta t_2 = k\Delta t_1$ where $k \in [0, 1]$ is randomly generated each time step. The results obtained using standard TVD RK-2 are also shown for the sake of comparison in the last two columns.

It then takes another forward Euler step to compute $\hat{\phi}^{n+3/2}$ from $\hat{\phi}^{n+1/2}$, and finally computes ϕ^{n+1} as

$$\phi^{n+1} = \frac{2}{3}\hat{\phi}^{n+3/2} + \frac{1}{3}\phi^n = \phi^n + \Delta t \left(\frac{\hat{\phi}^{n+3/2} - \phi^n}{3\Delta t/2} \right) \tag{19}$$

Although it is desirable to have a scheme similar to TVD RK-2 where all the Euler steps can be taken with arbitrary time step sizes in order to preserve positivity, we show in Appendix II that such a scheme is not practical or even necessarily feasible. Hence, we propose the following alternative scheme: The first Euler step is taken with a time step Δt_1 that is positivity preserving. If the time step restriction computed for the second Forward Euler step $\Delta t_2 \geq \Delta t_1$, then we set $\Delta t_2 = \Delta t_1$. Otherwise, if $\Delta t_2 < \Delta t_1$, we rewind ϕ back to ϕ^n and take the first Euler step with the time step $\Delta t_1/2$ (repeating if necessary). Similarly, for the third Euler step, if the computed time step $\Delta t_3 \geq \Delta t_1$, then we set $\Delta t_3 = \Delta t_1$. Otherwise, if $\Delta t_3 < \Delta t_1$, we rewind the state ϕ back to ϕ^n and take the first Euler step with the time step $\Delta t_1/2$ (again, repeating if necessary). The resulting scheme is the standard TVD RK-3 scheme with a time step that is positivity preserving for all three Euler steps. In scenarios where this time step becomes smaller than a desirable threshold, we do not clamp the time step further but instead clamp the fluxes as is discussed next in Section 5.

Case	A	B	C	Root	Robust Δt_{e_i}
1	$\gg 0$	$\gg 0$	ε	$C/2B + O(C^2)$	0
2	$\gg 0$	$\ll 0$	ε	inequality always true	Δt_{CFL}
3	$\ll 0$	$\gg 0$	ε	$C/2B + O(C^2)$	0
4	$\ll 0$	$\ll 0$	ε	$2B/A + \lvert O(C)\rvert$	$2B/A$
5	$\pm\varepsilon$	$\gg 0$	ε	$C/2B + O(AC^2)$	0
6	$\pm\varepsilon$	$\ll 0$	ε	inequality always true	Δt_{CFL}
7	$\gg 0$	$\pm\varepsilon$	ε	$O(B/A)$ or imaginary	0
8	$\ll 0$	$\pm\varepsilon$	ε	$\sqrt{-C/A} + O(B) + O(C^{1/2})$	0
9	$\pm\varepsilon$	$\pm\varepsilon$	ε	highly uncertain	0

Table 3: Limiting cases when C is a small positive number approaching zero. The Root column denotes the limiting value of the root $(B - \sqrt{B^2 - AC})/A$. The Robust Δt_{e_i} column denotes the numerically robust value that can be assigned to Δt_{e_i}.

5. Flux clamping

Here we consider the remaining limiting cases for roots of the inequality (7), which occur as C approaches zero. In these cases, either the internal energy e or the density ρ, or both, approach zero as can be seen from equation (11). An extremely small time step may be required when C approaches zero as shown in Cases 1 and 4 in Table 1. Table 3 enumerates all possible cases for A and B as C approaches zero. When $B \gg 0$, the root approaches $C/2B$ as shown in **Case 1**, **Case 3**, and **Case 5**, and there is no robust positive root. When $B \ll 0$, the inequality is either always true or the root approaches $2B/A$ from above as shown in **Case 2**, **Case 4**, and **Case 6**. **Case 7** has two sub-cases. Either $B^2 < AC$ and the root is imaginary, or the root is small and positive. Since it is not possible to distinguish between the two cases, we must set the robust root to zero. **Case 8** also requires a robust root of zero. Finally, when all three coefficients approach zero, the root is highly uncertain and the only robust root is zero.

In several cases, positivity dictates that the robust time step be driven to zero. As can be seen from equation (4) one could alternatively clamp the flux divided differences. However, this may violate conservation since fluxes affect multiple grid cells. Alternatively, one could clamp the fluxes at individual faces in order to maintain a larger time step; however, this results in a globally coupled problem as every flux affects its left and right cells. To remedy this, given the dimension d we make $2d$ co-located copies of the cell called "sub-cells", assign each sub-cell $1/(2d)$ of the mass, momentum, and energy, and associate each sub-cell with a unique flux face. This decouples the problem and allows the flux at each face to be clamped independently. We keep a global threshold Δt_g for the size of the minimum allowable time step, and clamp the fluxes whenever the time step size becomes less than Δt_g.

Assume that the density, momentum, and energy in sub-cell i are $\hat{\rho}_i^n = \rho_i^n/(2d)$, $(\widehat{\rho u})_i^n = (\rho u)_i^n/(2d)$ and $\hat{E}_i^n = E_i^n/(2d)$ respectively. Consider the density flux $\mathcal{D}_{i+1/2}^n$ at face $i + 1/2$. We clamp this flux such that $\hat{\rho}_i$ remains at least $\varepsilon_\rho \hat{\rho}_i^n$, i.e., we compute Δt such that

$$\hat{\rho}_i^n - \Delta t \mathcal{D}_{i+1/2}^n / \Delta x \geq \varepsilon_\rho \hat{\rho}_i^n. \tag{20}$$

This is done for both the left and right sub-cells obtaining Δt_{left} and Δt_{right} respectively. Then the flux $\mathcal{D}_{i+1/2}^n$ is scaled down by a factor of $\Delta t_{min}/\Delta t_g = \min(\Delta t_{right}, \Delta t_{left})/\Delta t_g$, guaranteeing positivity when we take the global time step Δt_g. Doing this for all $2d$ sub-cells guarantees that the parent cell has density at least $2d\varepsilon_\rho \hat{\rho}_i^n = \varepsilon_\rho \rho_i^n$ after the temporal update.

The internal energy at time t^{n+1} should not shrink below $\varepsilon_e \hat{e}_i^n$ yielding the inequality

$$\hat{e}_i^{n+1} = \frac{\hat{E}_i^{n+1}}{\hat{\rho}_i^{n+1}} - \frac{||(\widehat{\rho u})_i^{n+1}||^2}{2(\hat{\rho}_i^{n+1})^2} = \frac{\hat{E}_i^n - \Delta t \mathcal{E}_{i+1/2}^n/\Delta x}{\hat{\rho}_i^n - \Delta t \mathcal{D}_{i+1/2}^n/\Delta x} - \frac{||(\widehat{\rho u})_i^n - \Delta t \vec{\mathcal{M}}_{i+1/2}^n/\Delta x||^2}{2(\hat{\rho}_i^n - \Delta t \mathcal{D}_{i+1/2}^n/\Delta x)^2} \geq \varepsilon_e \hat{e}^n \tag{21}$$

which has a one to one correlation with inequality (6) after grouping various terms. Thus we can solve an equivalent inequality (7) to find a robust time step Δt. Similar to the density as described above, $\mathcal{D}_{i+1/2}^n$,

$\vec{\mathcal{M}}^n_{i+1/2}$ and $\mathcal{E}^n_{i+1/2}$ are then scaled down by a factor of $\Delta t_{min}/\Delta t_g$ if necessary. It is important to note that while solving inequality (21) via inequality (7), Tables 1 and 3 give the robust time step which is often identically zero. Unlike in Section 3 where this would have driven the time step to zero, here we are able to maintain $\Delta t \geq \Delta t_g$ and instead drive the flux to zero for the poorly behaved cases in Table 3. For our examples Δt_g was chosen as a fraction of the time step size dictated by applying the standard CFL restriction for the explicit scheme to the initial conditions, ensuring that the actual time steps had the same order of magnitude compared to those that the explicit scheme would have taken. This strategy worked well for all our examples because the velocity $\parallel |u| + c \parallel_\infty$ was highest initially and the flow smoothed out over time, making the initial conditions the most complex to resolve for positivity preservation. However, one could employ a different strategy for fluid flows that start slow and develop high speed non-linearities over time.

A straightforward approach would be to clamp all the fluxes using the minimum of the density and internal energy scaling factors, and this is indeed what we do when the internal energy scaling factor is more restrictive. However, when the density scaling factor is more restrictive, we first try to clamp the density flux only and recompute the scaling factor for internal energy using the newly clamped density flux in inequality (21). If the recomputed scaling factor for internal energy is less restrictive than the original internal energy scaling factor, then we further clamp all fluxes with the newly recomputed scaling factor. Otherwise, if the recomputed scaling factor for internal energy is more restrictive than the original one, we simply stick with the straightforward approach of using the minimum of the density and internal energy scaling factors.

6. Numerical results - explicit time integration

We have thoroughly tested a large number of examples as well as parameters, and report on a representative sampling of those tests here. We utilized both ENO-LLF and ENO-RF [38] and varied the order of accuracy of both the ENO scheme and the TVD-RK scheme between 1 and 3. Although our method is valid for any equation of state, we used the gamma law gas $p = (\gamma - 1)\rho e$, with $\gamma = 1.4$. The constants ε_ρ and ε_e were set to .5 in all examples. [44] proposed a positivity limiter for third order accurate ENO and third order accurate TVD-RK which guarantees positivity preservation under the time step restriction $(\Delta t/\Delta x) \parallel |u| + c \parallel_\infty \leq 1/6$. To compare with the time steps that would have been taken by their scheme, we also plot the curve $\Delta t = \Delta x/(6 \parallel |u| + c \parallel_\infty)$, but note that our plot may be slightly different from that of [44] because u and c vary according to the numerical method, truncation errors, time steps, and grid resolutions.

6.1. One-dimensional examples

The Sedov blast wave is a typical problem where a shock at the center of the domain drives the density to zero analytically [21, 36]. The computational domain is $[-2, 2]$. Initially, $\rho = 1$, $u = 0$, and $E = 10^{-12}$ everywhere except the center cell where $E = 3,200,000/\Delta x$. Figure 4 depicts the solution obtained using ENO-LLF-3, TVD-RK-1, and CFL=.5 with our adaptive time step restriction. Without our adaptive time step restriction this choice of scheme and parameters is unable to run to completion. Figure 4 (bottom right) compares our adaptive time step size to that taken by the scheme using ENO-LLF-1, TVD-RK-1, and CFL=.5, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction. Resolutions 6400 and 12800 also required flux clamping, and we used $\Delta t_g = 1 \times 10^{-9}$ and $\Delta t_g = 5 \times 10^{-10}$ respectively.

Next, consider the double rarefaction problem [23, 41] where a very low density is generated in the center of the domain. Initial conditions are $\rho_L = \rho_R = 7$, $u_L = -1$, $u_R = 1$, and $p_L = p_R = .2$, with the discontinuity at $x = 0$. Figure 5 depicts the solution obtained using ENO-RF-3, TVD-RK-2, and CFL=.6 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 5 (bottom right) compares our adaptive time step size to that taken by the scheme using ENO-RF-1, TVD-RK-2, and CFL=.6, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction.

Consider the Leblanc shock tube problem where the initial conditions are $\rho_L = 2$, $\rho_R = .001$, $u_L = u_R = 0$, $p_L = 10^9$, and $p_R = 1$, with the discontinuity at $x = 0$. Figure 6 depicts the solution obtained using

ENO-LLF-3, TVD-RK-3, and CFL=.7 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 6 (bottom right) compares our adaptive time step size to that taken by the scheme using ENO-LLF-1, TVD-RK-3, and CFL=.7, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction.

Consider the shock reflection problem at hypervelocities with initial conditions $\rho_L = \rho_R = 1$, $u_L = 3000$, $u_R = 1000$, and $p_L = p_R = 10^4$, with the discontinuity at $x = 0$. Figure 7 depicts the solution obtained using ENO-LLF-3, TVD-RK-2, and CFL=.8 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 7 (bottom right) compares our adaptive time step size to that taken by the scheme using ENO-LLF-1, TVD-RK-2, and CFL=.8, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction. Resolutions 6400 and 12800 also required flux clamping, and we used $\Delta t_g = 1 \times 10^{-8}$ and $\Delta t_g = 5 \times 10^{-9}$ respectively.

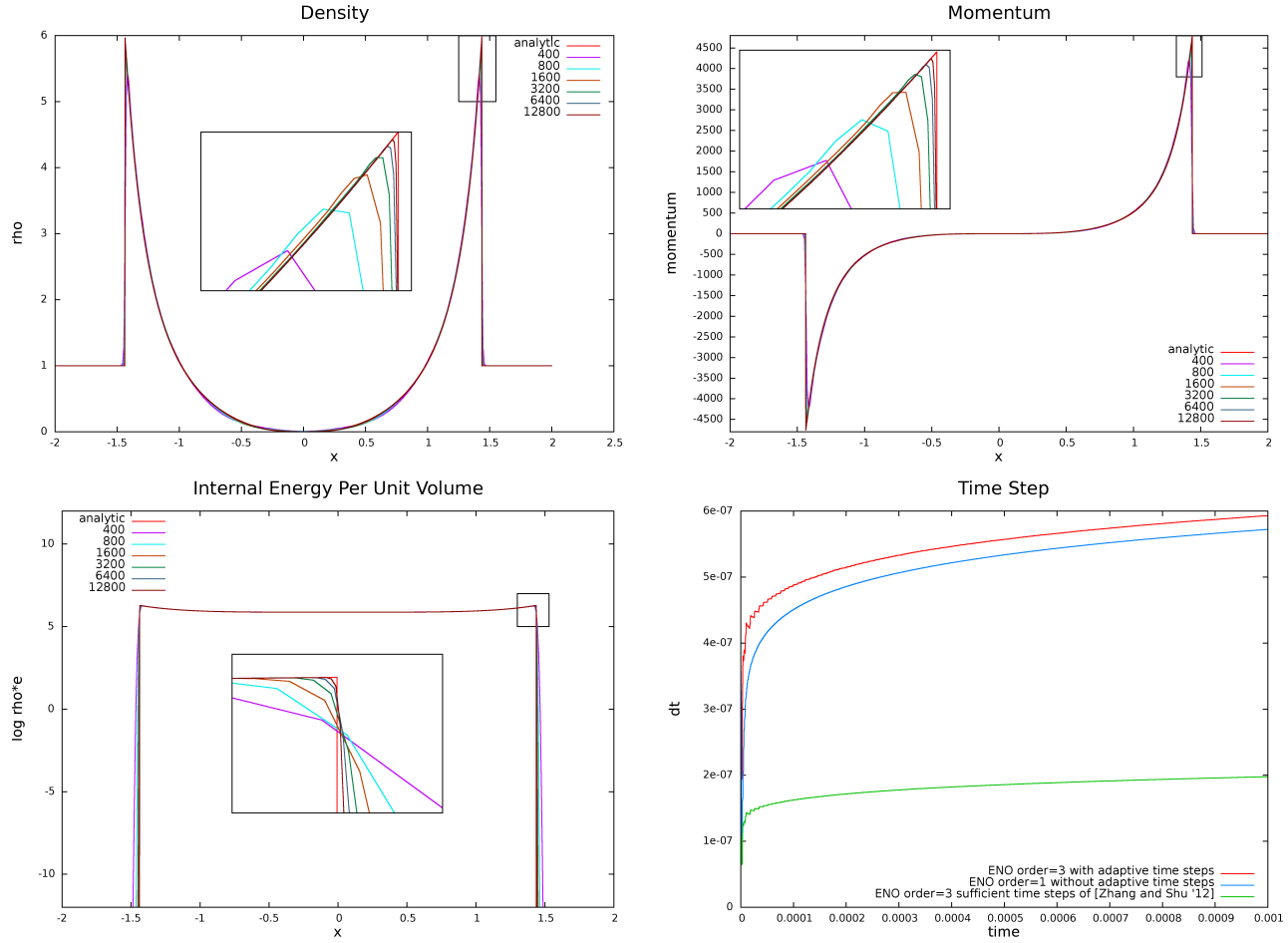


Figure 4: Numerical profiles for the one dimensional Sedov blast wave problem at $t = .001$ using ENO-LLF-3, TVD-RK-1, and CFL=.5 with our adaptive time step restriction. The profiles converge to the analytic solution (shown in red) under grid refinement.

Finally, consider the vacuum generation problem at hypervelocities [26, 27]. Initial conditions are $\rho_L = \rho_R = .35$, $u_L = 1000$, $u_R = 3000$, and $p_L = p_R = 10^4$. Figure 8 depicts the solution obtained using ENO-LLF-3, TVD-RK-2, and CFL=.75 with our adaptive time step restriction. This choice of scheme and

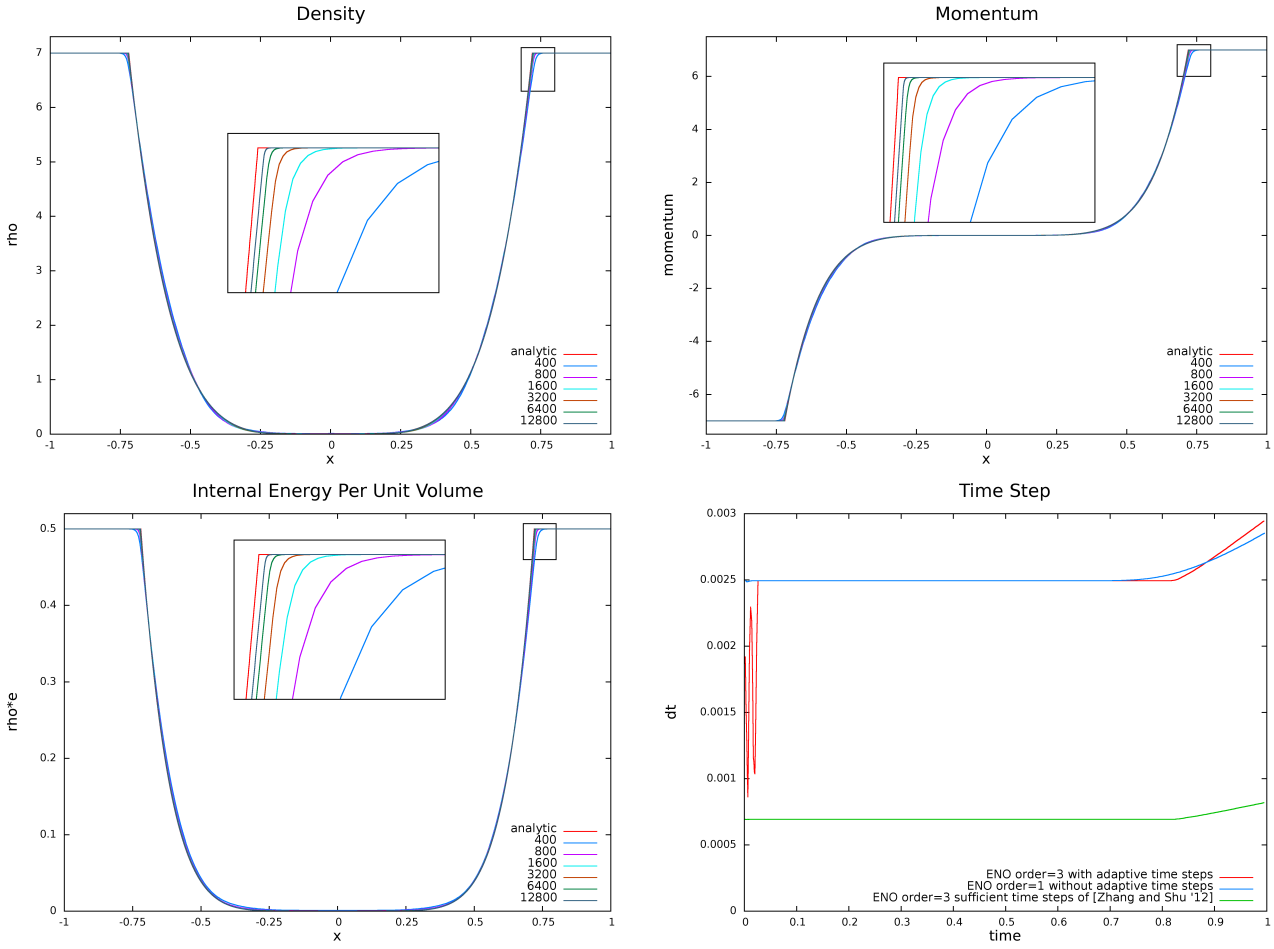


Figure 5: Numerical profiles for the one-dimensional double rarefaction problem at $t = .6$ using ENO-RF-3, TVD-RK-2, and CFL=.6 with our adaptive time step restriction. The profiles converge to the analytic solution (shown in red) under grid refinement.

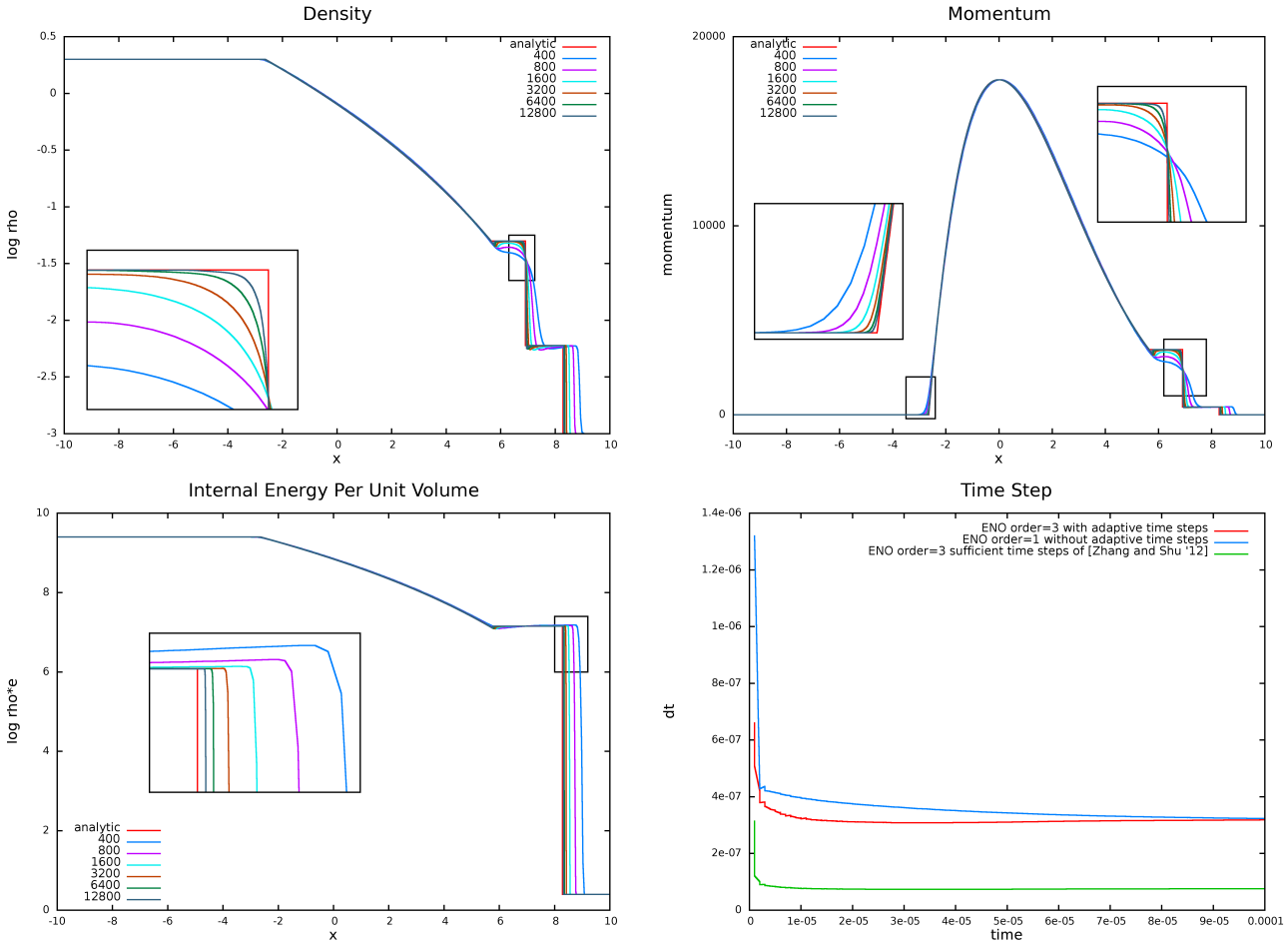


Figure 6: Numerical profiles for the 1D Leblanc shock tube at $t = .0001$ using ENO-LLF-3, TVD-RK-3, and CFL=.7 with our adaptive time step restriction. The results converge to the analytic solution (shown in red) under grid refinement. Note that the density and internal energy were plotted on a log scale so that the interesting features were not inordinately compressed.

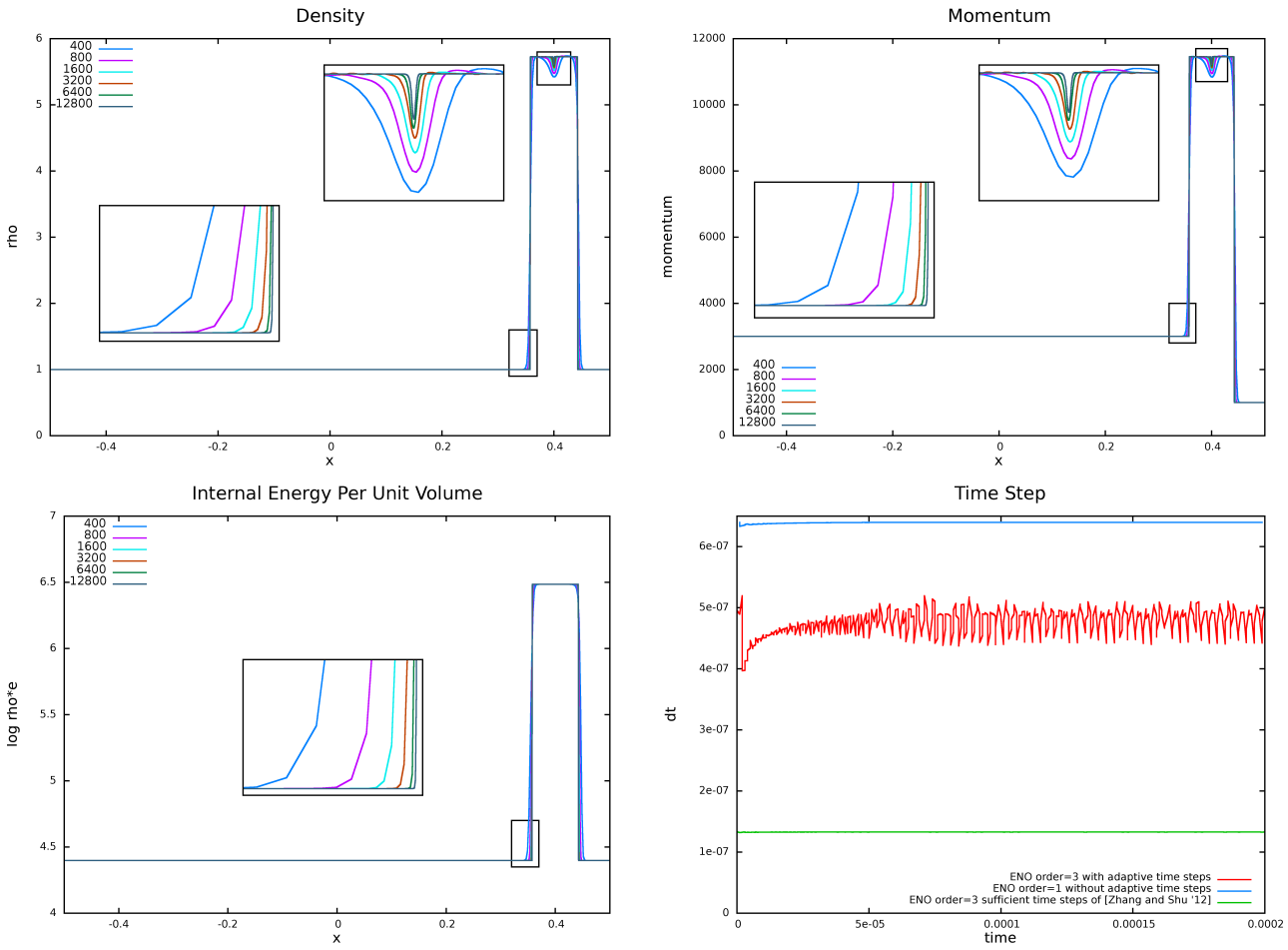


Figure 7: Numerical profiles for the 1D shock reflection problem at hypervelocities at $t = .0002$ using ENO-LLF-3, TVD-RK-2, and CFL=.8 with our adaptive time step restriction.

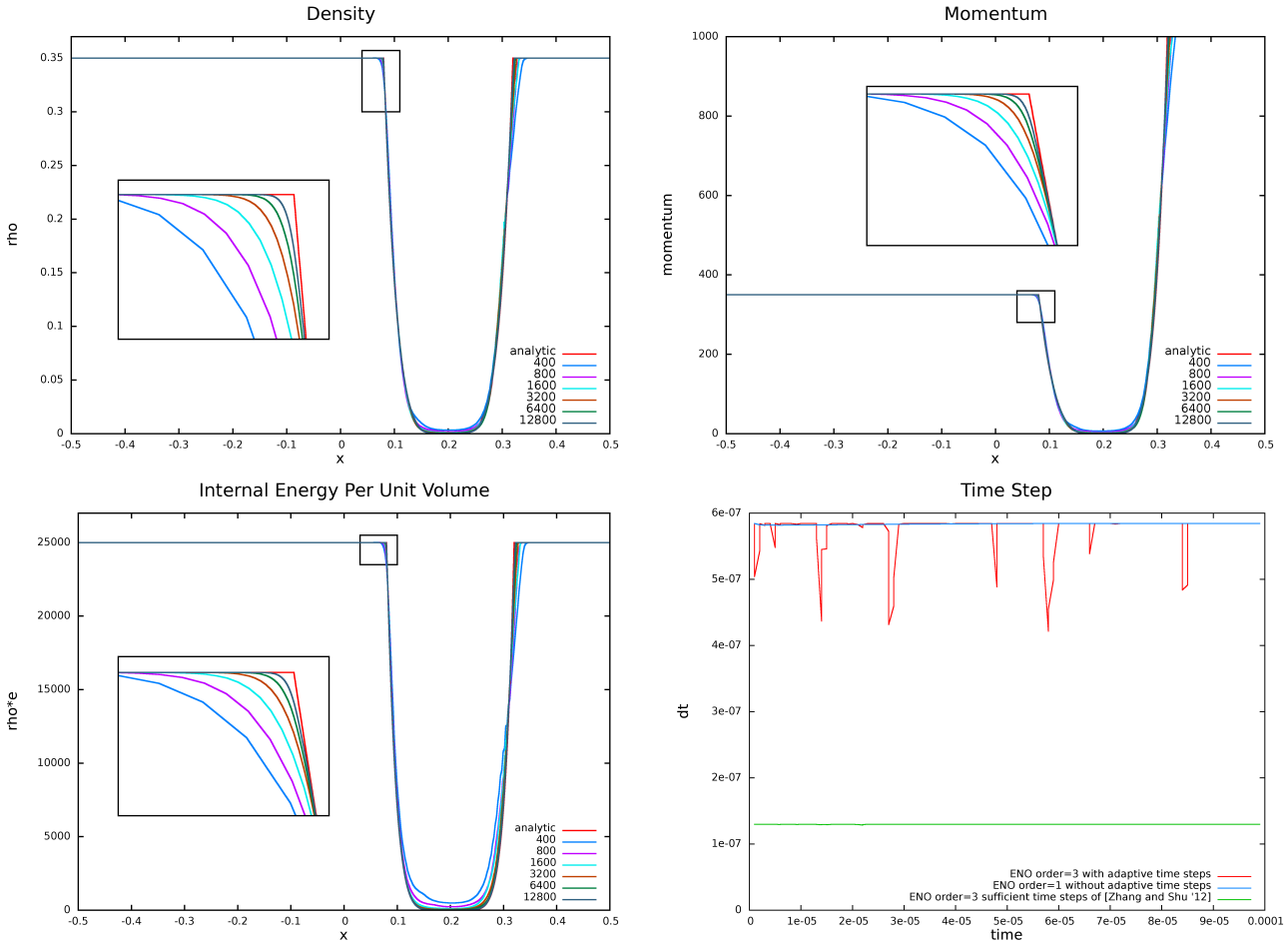


Figure 8: Numerical profiles for the 1D vacuum generation problem at hypervelocities at $t = .0001$ using ENO-LLF-3, TVD-RK-2, and CFL=.75 with our adaptive time step restriction. The results converge to the analytic solution (shown in red) under grid refinement.

parameters is unable to run to completion without our adaptive time step restriction. Figure 8 (bottom right) compares our adaptive time step size to that taken by the scheme using ENO-LLF-1, TVD-RK-2, and CFL=.75, which is the selection of parameters that allows the code to run to completion without adaptive time step restriction. Resolutions 3200, 6400, and 12800 also required flux clamping, and we used $\Delta t_g = 1 \times 10^{-8}$, $\Delta t_g = 5 \times 10^{-9}$, and $\Delta t_g = 2.5 \times 10^{-9}$ respectively.

6.2. Two-dimensional examples

We simulated a kinematic square block with side length .4 initially centered at $x = .2$ and moving in the positive x-direction with a speed of 5. The computational domain is $[-5, 5] \times [-5, 5]$, and initially $\rho = 1$, $u = 0$, and $p = 1$ everywhere. Figure 9(left) shows 30 equally spaced density contours between $\rho = 0$ and $\rho = 5$ at $t = .8$ obtained using ENO-LLF-2, TVD-RK-3, and CFL=.5 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 9(right) shows the density contour of $\rho = 1.25$ at $t = .8$ at various resolutions to illustrate convergence under grid refinement. The black contour shows the ground truth achieved by the scheme using ENO-LLF-3, TVD-RK-1, and CFL=.5, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction.

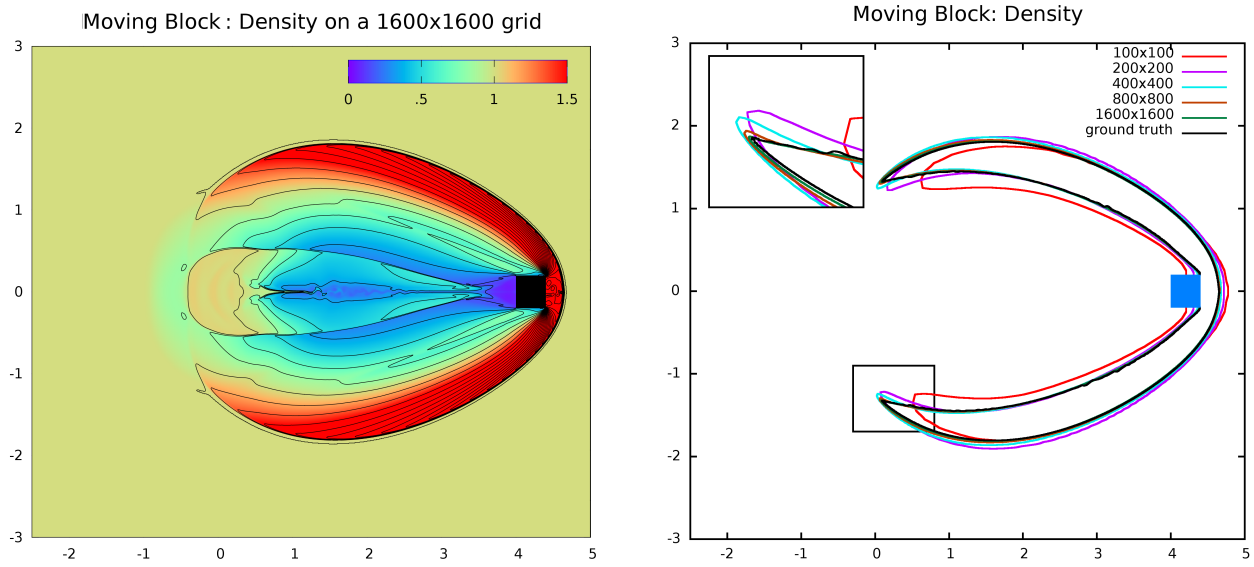


Figure 9: The kinematic block moving in the positive x-direction (left) 30 equally spaced density contours between $\rho = 0$ and $\rho = 5$ at $t = .8$. Note that the color map only goes from $\rho = 0$ to $\rho = 1.5$ to accentuate the details behind the block. (right) Density contour of $\rho = 1.25$ at $t = .8$ at various resolutions to illustrate convergence under grid refinement.

Consider a shock diffracting over a backward facing corner as in [6, 41, 44]. The computational domain is $[0, 13] \times [0, 11]$ with a corner given by $[0, 1] \times [0, 6]$. The initial condition is a pure right-moving shock of Mach number 5.09 initially located at $x = .5$. The air in front of the shock is at rest with $\rho = 1.4$ and $p = 1$. The boundary conditions are inflow at $x = 0$ and reflective everywhere else. Figure 10(left) shows 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 2.3$ obtained using ENO-LLF-3, TVD-RK-3, and CFL=.7 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 10(right) shows the density contour of $\rho = 1.5$ at various grid resolutions to illustrate convergence under grid refinement. Note that the results are similar to those in the literature [6, 41, 44]. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-3}$ for resolution 130×110 and successively halved Δt_g each time the resolution was doubled.

Consider a channel with a computational domain of $[0, 2] \times [0, .5]$, where the bottom has three solid humps defined by $y = .2 \sin(3\pi x)$, similar to [27]. We make higher humps to create a more difficult problem forcing

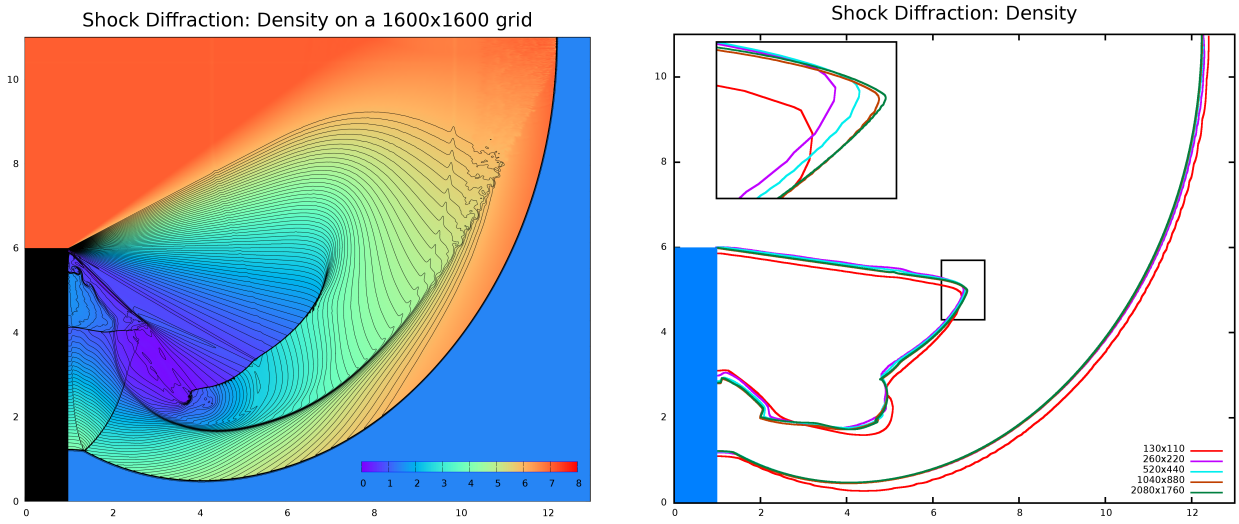


Figure 10: The shock diffraction problem where a shock passes a backward facing corner [6, 41, 44] (left) 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 2.3$. (right) Density contour of $\rho = 1.5$ at $t = 2.3$ at various resolutions to illustrate convergence under grid refinement.

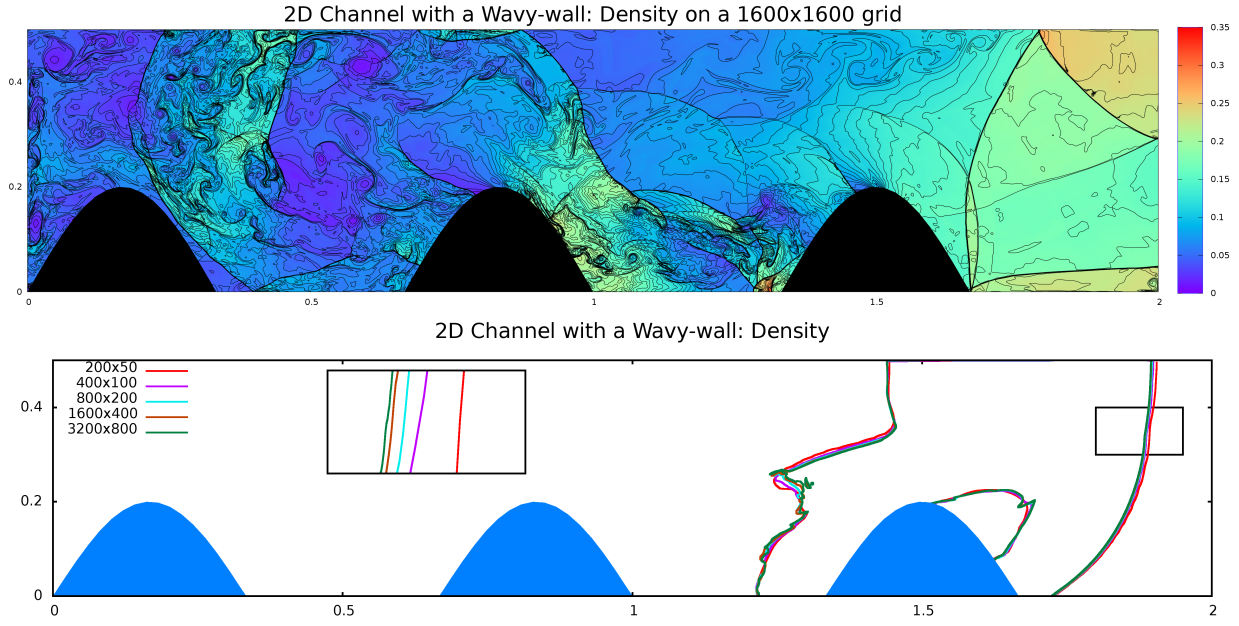


Figure 11: A wavy wall channel similar to [27]. (top) 40 equally spaced density contours between $\rho = 0$ and $\rho = .3$ at $t = .0036$. (bottom) Density contour of $\rho = .16$ at $t = .0011$ at various resolutions to illustrate convergence under grid refinement.

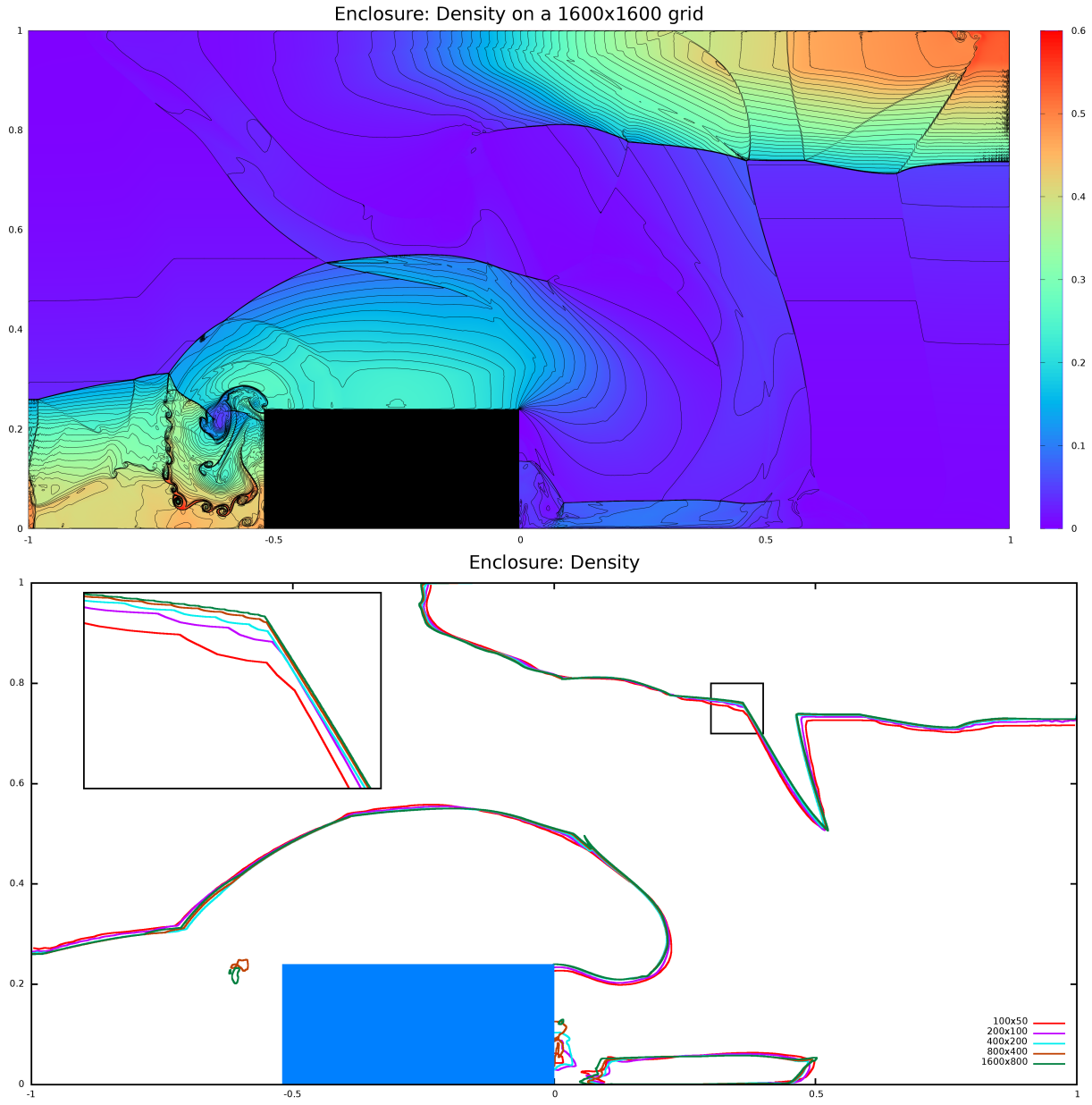


Figure 12: The two dimensional enclosure problem [27]. (top) 60 equally spaced density contours from $\rho = 0$ to $\rho = 6$ at $t = .00075$. (bottom) Density contour of $\rho = .05$ at $t = .00075$ at various grid resolutions to illustrate convergence under grid refinement.

a wider range of parameters to not run to completion. Initial conditions are $p = 10^4$, $\rho = .1$, and $v = 0$ everywhere. If $x \leq 1$, then $u = 100$, otherwise $u = 10$. The boundary conditions are reflective everywhere. Figure 11(top) shows 40 equally spaced density contours between $\rho = 0$ and $\rho = .3$ at $t = .0036$ obtained using ENO-LLF-3, TVD-RK-2, and CFL=.5 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 11(bottom) shows the density contour of $\rho = .16$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-6}$ for scale 200×50 and successively halved Δt_g each time the resolution was doubled.

Consider the enclosure problem [27] with computational domain of $[-1, 1] \times [0, 1]$, with a block located within $[-.52, 0] \times [0, .24]$. Initial conditions are $p = 10^4$, $\rho = .1$, and $u = 100$ everywhere. If $x \leq 0$, then $v = -1000$, otherwise $v = 1000$. Figure 12(top) shows 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = .00075$ obtained using ENO-LLF-3, TVD-RK-3, and CFL=.5 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 12(bottom) shows the density contour of $\rho = .05$ at $t = .00075$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 5 \times 10^{-7}$ for resolution 100×50 and successively halved Δt_g each time the resolution was doubled.

7. Semi-implicit time integration

We follow the semi-implicit framework of [22] where the flux vector was split into an advection part and a non-advection part

$$\mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} \rho\vec{u} \\ \rho\vec{u} \otimes \vec{u} \\ E\vec{u} \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} 0 \\ p \\ p\vec{u} \end{pmatrix} \tag{22}$$

The advection part is integrated explicitly to obtain intermediate values $\rho^\star$, $(\rho\vec{u})^\star$, and $E^\star$. Since the continuity equation is independent of the pressure $\rho^{n+1} = \rho^\star$. Note that we utilize the adaptive time step restriction and flux clamping techniques as outlined in Sections 3 and 5, to ensure that this update is positivity preserving. It is straightforward to use TVD-RK in order to improve the efficacy of the advection-only step.

The non-advection momentum and energy updates are

$$\frac{(\rho\vec{u})^{n+1} - (\rho\vec{u})^\star}{\Delta t} = -\nabla p, \quad \frac{E^{n+1} - E^\star}{\Delta t} = -\nabla \cdot (p\vec{u}) \tag{23}$$

Dividing the momentum update equation by ρ^{n+1} results in

$$\vec{u}^{n+1} = \vec{u}^\star - \Delta t \frac{\nabla p^{n+1}}{\rho^{n+1}} \tag{24}$$

and taking the divergence gives

$$\nabla \cdot \vec{u}^{n+1} = \nabla \cdot \vec{u}^\star - \Delta t \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho^{n+1}} \right) \tag{25}$$

Then the pressure evolution equation [9]

$$p_t + \vec{u} \cdot \nabla p = -\rho c^2 \nabla \cdot \vec{u} \tag{26}$$

is semi-discretized by fixing $\nabla \cdot \vec{u}$ to time t^{n+1}, and the equation of state is used to compute the advected pressure $p_a = p^\star = p(\rho^\star, e^\star)$ as in [13]. Substituting $p^\star$ into the semi-discretized form of equation (26) gives

$$p^{n+1} = p^\star - \Delta t \rho c^2 \nabla \cdot \vec{u}^{n+1} \tag{27}$$

Finally, combining equations (25) and (27) results in

$$p^{n+1} - \Delta t^2 \rho^n (c^2)^n \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho^{n+1}} \right) = p^\star - \Delta t \rho^n (c^2)^n \nabla \cdot \vec{u}^\star \tag{28}$$

where the term ρc^2 has been fixed to time t^n. The $\rho^n (c^2)^n$ terms are assembled into a diagonal matrix $P = [\Delta t^2 \rho^n (c^2)^n]$ and the gradient and divergence operators are discretized to obtain the following system of equations

$$[P^{-1} + \nabla^T (\rho_f^{n+1})^{-1} \nabla] \tilde{p}^{n+1} = P^{-1} \tilde{p}^\star + \nabla^T \vec{u}_f^\star \tag{29}$$

where ∇ now denotes the discretized gradient operator and $-\nabla^T$ denotes the corresponding discretized divergence operator. Here $\tilde{p} = p \Delta t$ and

$$(\rho_f^{n+1})_{i+1/2} = \frac{\rho_i^{n+1} + \rho_{i+1}^{n+1}}{2}, \quad (\vec{u}_f^\star)_{i+1/2} = \frac{(\rho \vec{u})_i^\star + (\rho \vec{u})_{i+1}^\star}{\rho_i^{n+1} + \rho_{i+1}^{n+1}} \tag{30}$$

Note that the identity term (P^{-1}) in equation (29) allows for the efficient solution of cell-centered pressure values using fast solvers such as preconditioned conjugate gradient (PCG). Subsequently, the post advection (time t^*) state is updated in a conservative flux-based manner via the flux $\mathbf{F}_2(U) = (0, p_f, p_f \vec{u})^T$. To construct $\mathbf{F}_2$, face pressures are computed via

$$(\tilde{p}_f^{n+1})_{i+1/2} = \frac{\rho_i^{n+1} \tilde{p}_{i+1}^{n+1} + \rho_{i+1}^{n+1} \tilde{p}_i^{n+1}}{\rho_i^{n+1} + \rho_{i+1}^{n+1}} \tag{31}$$

and face velocities are computed by rewriting equation (24) using face-averaged quantities

$$\vec{u}_f^{n+1} = \vec{u}_f^\star - (\rho_f^{n+1})^{-1} \nabla \tilde{p}^{n+1} \tag{32}$$

Similar to equation (4) the flux-based implicit update then takes the form

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^* - \Delta t \frac{\mathbf{F}_2(U)_{i+1/2} - \mathbf{F}_2(U)_{i-1/2}}{\Delta x} \tag{33}$$

It is rather complicated to maintain positivity when dealing with the non-advection fluxes since they were solved for implicitly. Our aforementioned strategy which was designed to include the ability to deal with arbitrary fluxes becomes quite useful in such a situation. For the sake of exposition, algorithm 1 demonstrates the pseudo code of our modified approach for handling semi-implicit compressible flow (for TVD RK-2). First we store/cache the time t^n state (step 2). Then, as mentioned above, we use our adaptive time step restriction for updating the state $\mathbf{U}^n$ to $\mathbf{U}^\star$ with the advection fluxes $\mathbf{F}_1(\mathbf{U})$ and the time step Δt_{adv} (steps 4 through 7). Then, we use Δt_{adv} to implicitly solve for the non-advection fluxes $\mathbf{F}_2(\mathbf{U})$ (steps 8 and 9). Next, we compute the total flux $\mathbf{F}_1(\mathbf{U}) + \mathbf{F}_2(\mathbf{U})$ (step 10) and restore the current state back to its cached version, i.e. $\mathbf{U}^n$. We then use our adaptive time step restriction and flux clamping technique on the state $\mathbf{U}^n$ where the total flux is assumed to be $\mathbf{F}_1(\mathbf{U}) + \mathbf{F}_2(\mathbf{U})$, thereby ensuring positivity preservation (steps 11 through 17). Note that this requires the fluxes and not the flux divided differences. Also note that computing the effective advection fluxes $\mathbf{F}_1(\mathbf{U})$ requires the proper averaging of the individual fluxes from each TVD-RK step (steps 4 through 6). For example, for TVD-RK-2, the linearity of equation (16) allows us to write the effective advection flux as

$$\mathbf{F}_1(\mathbf{U}) = \frac{\Delta t_1 \mathbf{F}_{11}(\mathbf{U}) + \Delta t_2 \mathbf{F}_{12}(\mathbf{U})}{\Delta t_1 + \Delta t_2}. \tag{34}$$

where $\mathbf{F}_{11}(\mathbf{U})$ and $\mathbf{F}_{12}(\mathbf{U})$ are the fluxes in each of the two Euler steps. The effective flux for TVD RK-3 is computed similarly.

Algorithm 1 Simulation Loop

while $time < t_{target}$ **do**
  Compute the time step size Δt.
  $U_{save} = U^n$.
  Update U^n with a positivity preserving forward Euler step of size Δt_1 and fluxes $\mathbf{F}_{11}$.
  Take another positivity preserving forward Euler step of size Δt_2 and fluxes $\mathbf{F}_{12}$.
  Compute the effective advection flux $\mathbf{F}_1(\mathbf{U})$ as described in equation (34).
  Compute $\Delta t_{\mathsf{adv}} = \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}$.
  Solve the system in equation (29) with $\Delta t = \Delta t_{\mathsf{adv}}$ to obtain the pressure p.
  Use p along with the updated face velocities to construct the projection flux $\mathbf{F}_2(\mathbf{U}) = (0, p_f, p_f \vec{u}_f)^T$.
  Compute the effective flux $\mathbf{F}_{\mathsf{eff}} = \mathbf{F}_1(\mathbf{U}) + \mathbf{F}_2(\mathbf{U})$.
  Use adaptive time step restriction on $U = U_{save}$ with fluxes $\mathbf{F}_{\mathsf{eff}}$ to obtain $\Delta t_{\mathsf{final}}$.
  if $\Delta t_{\mathsf{final}} < \Delta t_g$ **then**
    $\Delta t_{\mathsf{final}} = \Delta t_g$
    Perform flux clamping on $\mathbf{F}_{\mathsf{eff}}$ to obtain $\hat{\mathbf{F}}_{\mathsf{eff}}$
  end if
  Update U_{save} to U^{n+1} with time step $\Delta t_{\mathsf{final}}$ and fluxes $\hat{\mathbf{F}}_{\mathsf{eff}}$ similar to equation (33)
  $time$ +=$\Delta t_{\mathsf{final}}$.
end while

7.1. Moving Objects

Throughout the paper, static objects are handled by filling ghost cells inside the solid using standard reflective boundary conditions. In the case of moving objects, one should advance the rigid bodies to time t^{n+1} after/during advection but before solving the implicit system for the pressure. Unfortunately, this does not work well here because $\mathbf{F}_2(\mathbf{U})$ is required to determine the size of the time step implying that we do not know the final location of the object prior to computing $\mathbf{F}_2(\mathbf{U})$. Thus, we use time-splitting alternatively updating the compressible flow and advancing the object, i.e., at the end of the time step we fill ghost cells inside the object, advance the object's position, and finally fill ghost cells inside the object again to prepare for the next advection step.

7.2. One-dimensional experiments

For the Sedov blast wave problem, Figure 13 depicts the solution obtained using ENO-LLF-3, TVD-RK-2, and CFL=.5 with our adaptive time step restriction for semi-implicit time integration. Without our adaptive time step restriction this choice of scheme and parameters is unable to run to completion. This example required flux clamping, and we used $\Delta t_g = 10^{-7}$ for resolution 400 and successively halved Δt_g each time the resolution was doubled. Figure 13 (bottom right) shows the time and location where flux clamping occurred.

In addition, Figure 14 compares our adaptive time step size to that taken by the scheme using ENO-LLF-1, TVD RK-2, and CFL=.5, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction. We also show the time steps sufficient for maintaining positivity using the method of [44]. Note that our method does not severely restrict the large time steps allowed by the semi-implicit scheme, and the time steps keep becoming larger as the flow smooths out over time.

For the double rarefaction problem, Figure 15 depicts the solution obtained using ENO-LLF-3, TVD-RK-3, and CFL=.5 with our adaptive time step restriction for semi-implicit time integration. Without our adaptive time step restriction this choice of scheme and parameters is unable to run to completion. This example required flux clamping, and we used $\Delta t_g = 10^{-3}$ for resolution 400 and successively halved Δt_g each time the resolution was doubled. Figure 15 (bottom right) shows the time and location where flux clamping occurred.

For the shock reflection problem at hypervelocities, Figure 16 depicts the solution obtained using ENO-LLF-3, TVD-RK-2, and CFL=.6 with our adaptive time step restriction for semi-implicit time integration.

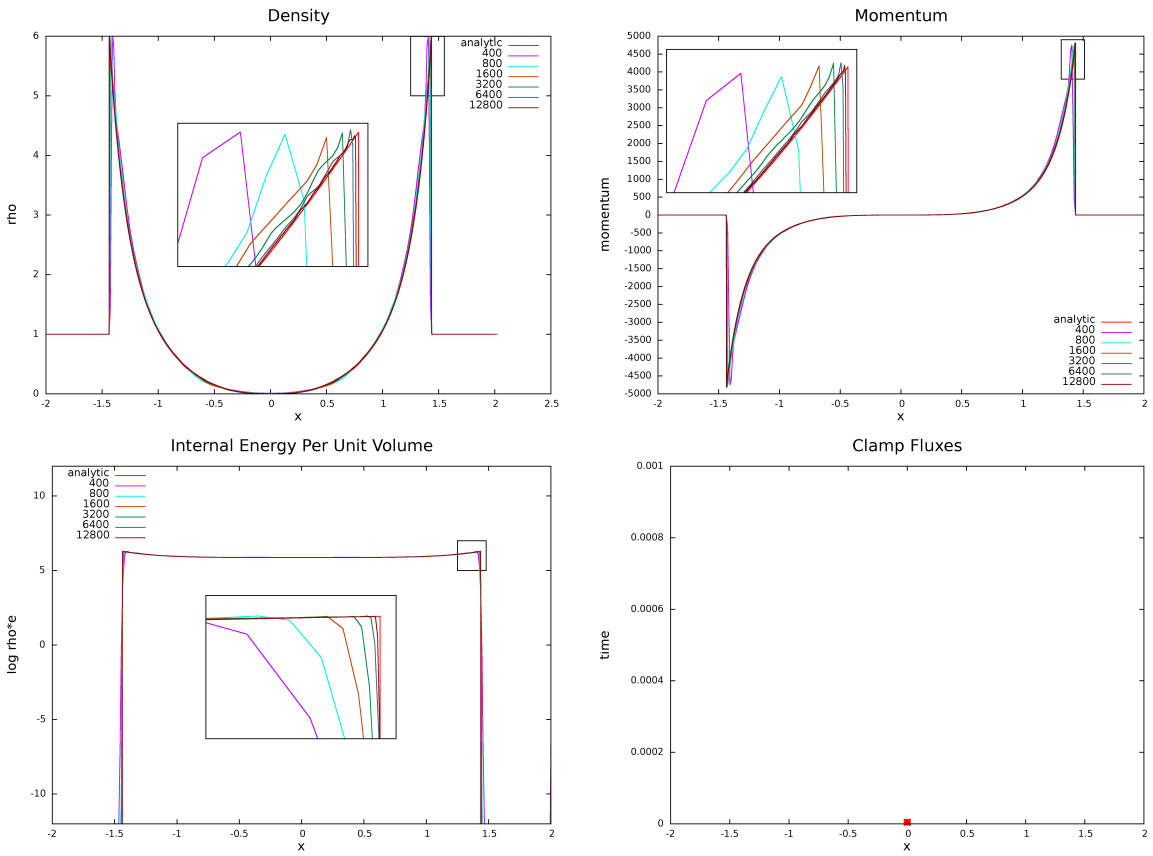


Figure 13: Numerical profiles for the one dimensional Sedov blast wave problem at $t = .001$ using ENO-LLF-3, TVD-RK-2, and CFL=.5 with our adaptive time step restriction. The profiles converge to the analytic solution (shown in red) under grid refinement.

Without our adaptive time step restriction this choice of scheme and parameters is unable to run to completion. Resolutions 3200, 6400, and 12800 also required flux clamping, and we used $\Delta t_g = 10^{-8}$, $\Delta t_g = 5\times10^{-9}$, and $\Delta t_g = 2.5\times10^{-9}$ respectively. Figure 16 (bottom right) shows the time and location where flux clamping occurred.

Remark: This example encountered a case in Table 3 where the robust time step was 0 when our adaptive time step restriction was used without flux clamping. However, when flux clamping was used with $\Delta t = \Delta t_g$, a positive density and internal energy was obtained without the need for any actual clamping - which is why Figure 16 (bottom right) does not show any clamped fluxes. In other words, our adaptive time step restriction is sometimes already sufficient as long as Δt is set to Δt_g.

For the Leblanc shock tube problem, Figure 17 depicts the solution obtained using ENO-LLF-3, TVD-RK-2, and CFL=.65 with our adaptive time step restriction for semi-implicit time integration. Without our adaptive time step restriction this choice of scheme and parameters is unable to run to completion. Resolutions 3200, 6400, and 12800 also required flux clamping, and we used $\Delta t_g = 10^{-8}$, $\Delta t_g = 5 \times 10^{-9}$, and $\Delta t_g = 2.5\times10^{-9}$ respectively. Figure 17 (bottom right) shows the time and location where flux clamping occurred.

7.3. Two-dimensional experiments

We simulated the two-dimensional Sedov blast wave problem for which the computational domain is a square. Initially density is 1, velocity is zero, and total energy is 10^{-12} everywhere except for the lower

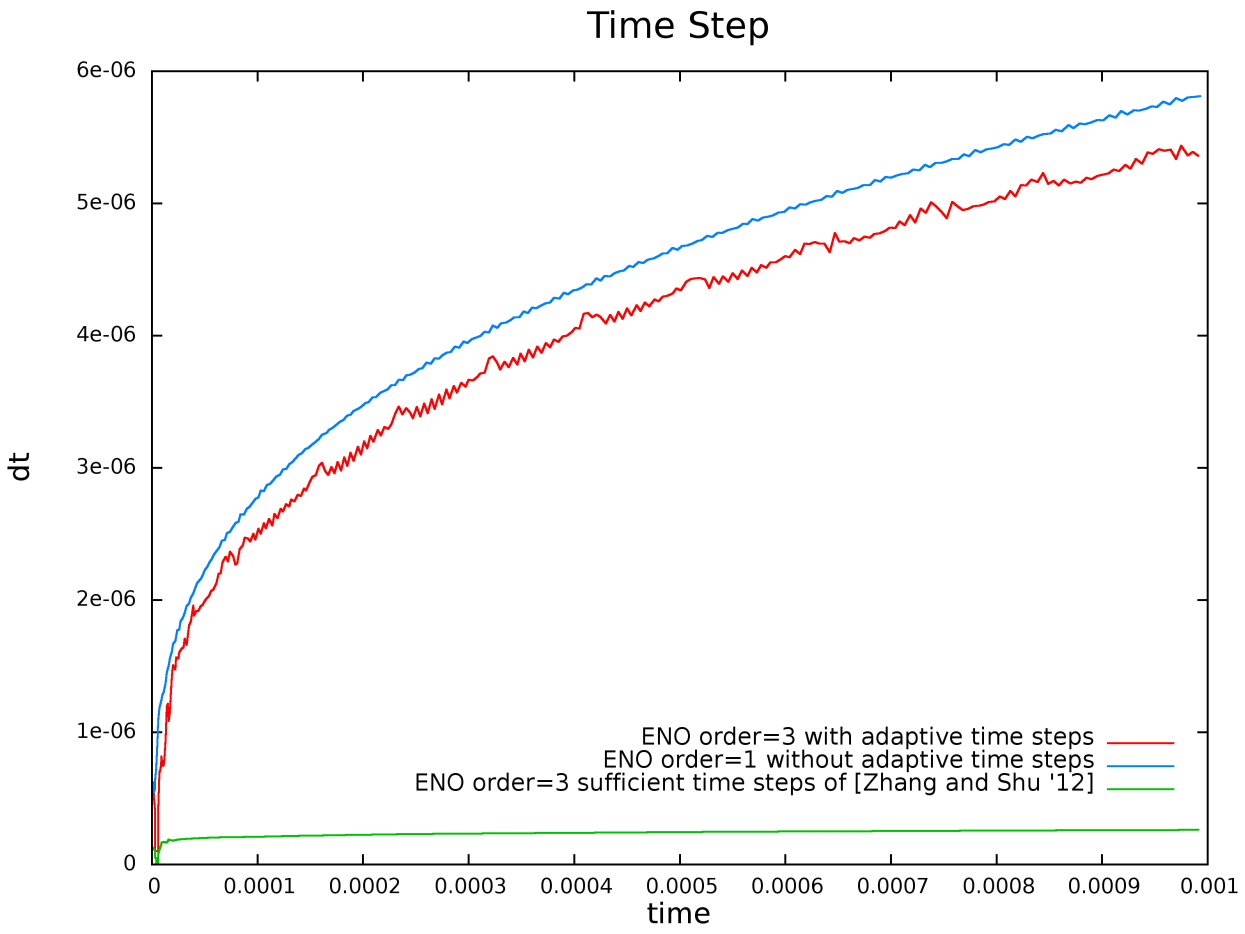


Figure 14: Time steps taken using our adaptive time step restriction with ENO-LLF-3, TVD-RK-2 and CFL=.5 for the Sedov blast wave problem (red), while those taken with ENO-LLF-1, TVD-RK-2 and CFL=.5 are shown in blue (which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction). Time steps sufficient for maintaining positivity using the method of [44] are shown in green.

left corner cell where it is the constant $\frac{.244816}{\Delta x \Delta y}$. The left and bottom edge of the domain have reflecting boundary conditions. Figure 18(left) shows 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 1$ obtained using ENO-LLF-3, TVD-RK-2, and CFL=.6 with our adaptive time step restriction for semi-implicit time integration. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 18(right) shows the density contour of $\rho = 4$ at various grid resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-4}$ for resolution 100×100 and successively halved Δt_g each time the resolution was doubled. Numerical results, shown in Figure 18, are comparable to those in the literature [24, 41].

For the moving block problem, Figure 19(left) shows 30 equally spaced density contours between $\rho = 0$ and $\rho = 5$ at $t = .8$ obtained using ENO-LLF-3, TVD-RK-3, and CFL=.5 with our adaptive time step restriction for semi-implicit time integration. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 19(right) shows the density contour of $\rho = 1.25$ at $t = .8$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-3}$ for resolution 100×100 and successively halved Δt_g each time the resolution was doubled. The black contour shows the ground truth computed by the explicit scheme using ENO-LLF-

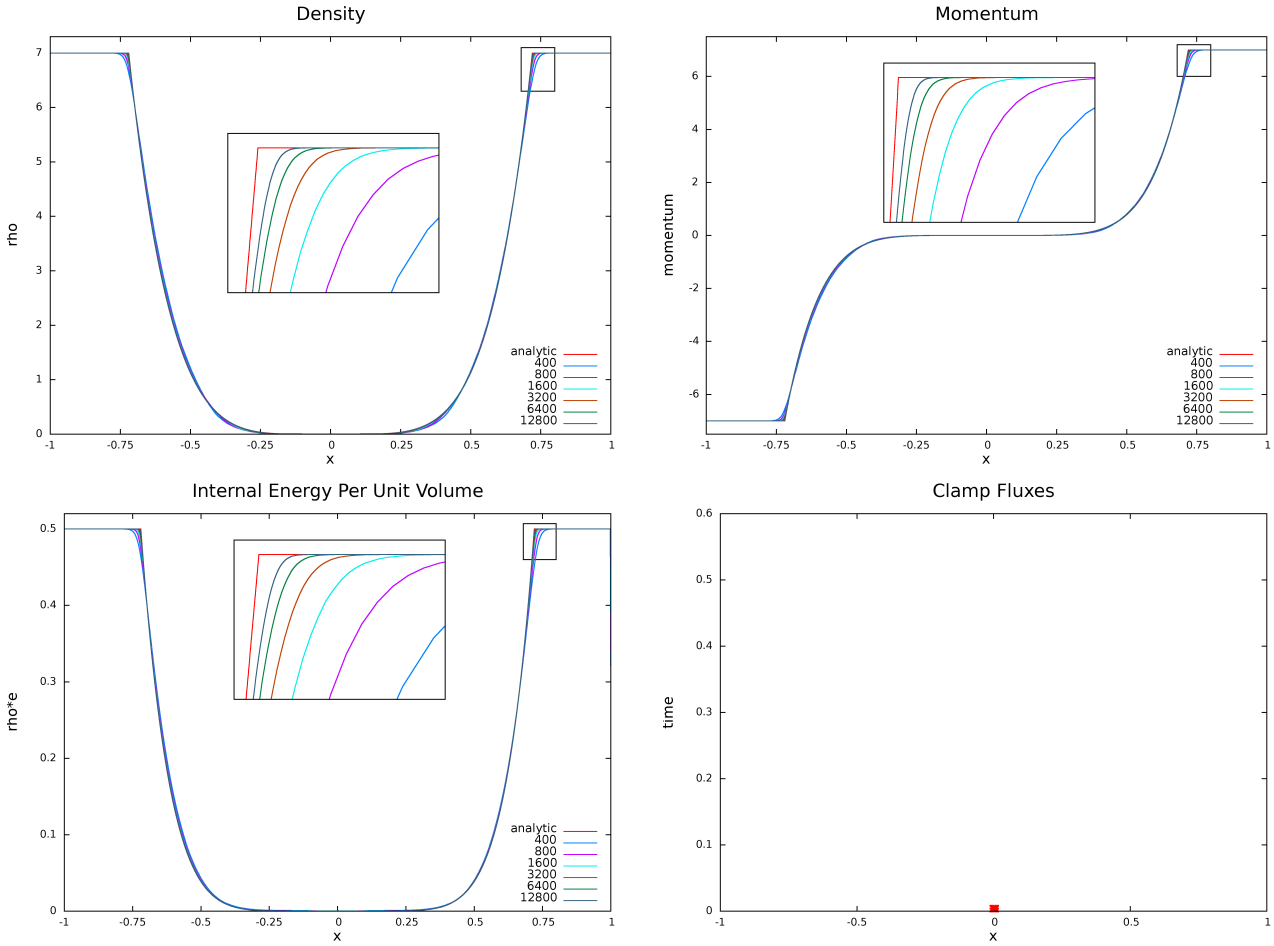


Figure 15: Numerical profiles for the one-dimensional double rarefaction problem at $t = .6$ using ENO-LLF-3, TVD-RK-3, and CFL=.5 with our adaptive time step restriction. The profiles converge to the analytic solution (shown in red) under grid refinement.

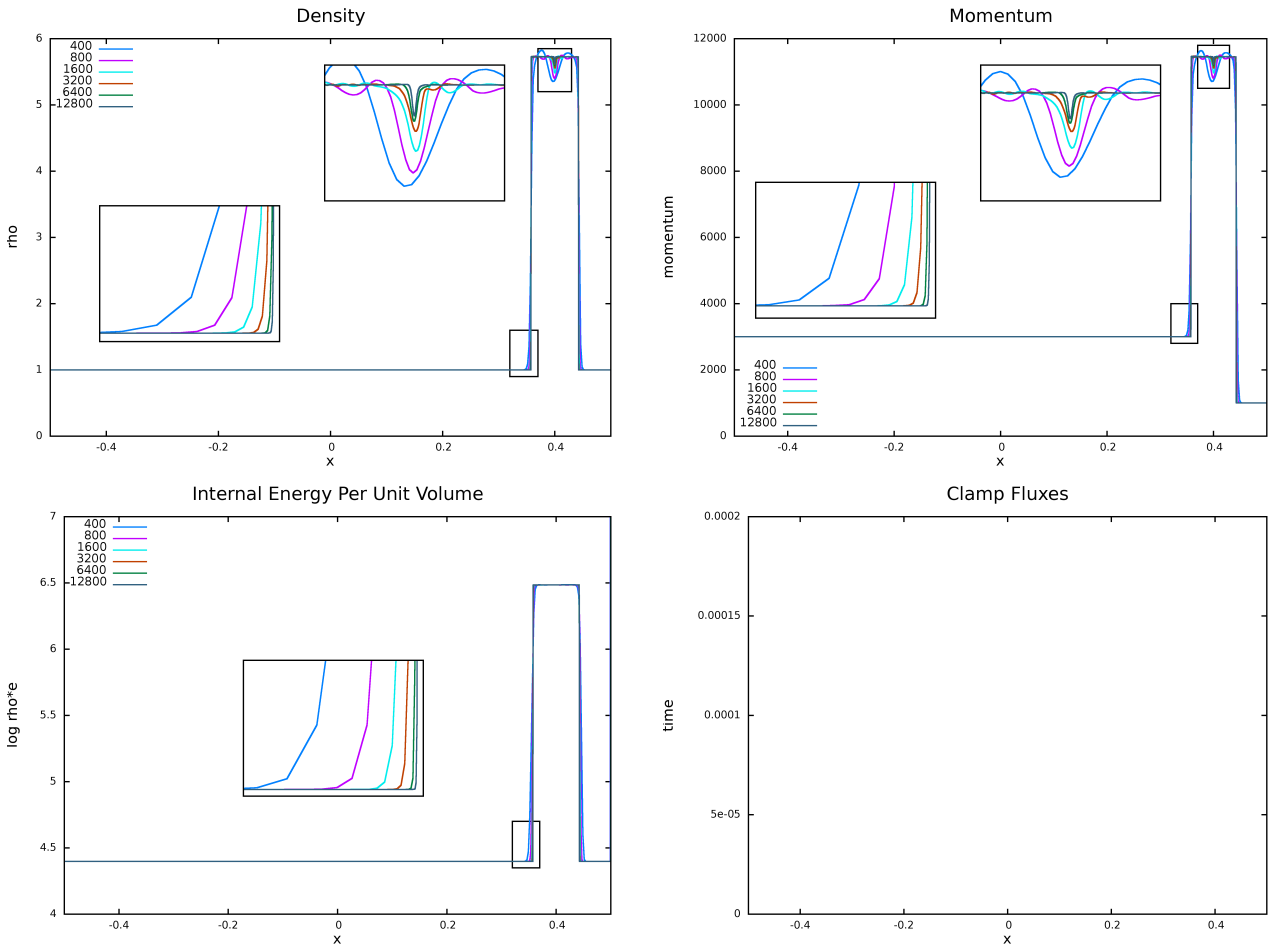


Figure 16: Numerical profiles for the shock reflection problem at hypervelocities at $t = .0002$ using ENO-LLF-3, TVD-RK-2, and CFL=.6 with our adaptive time step restriction. The profiles converge under grid refinement.

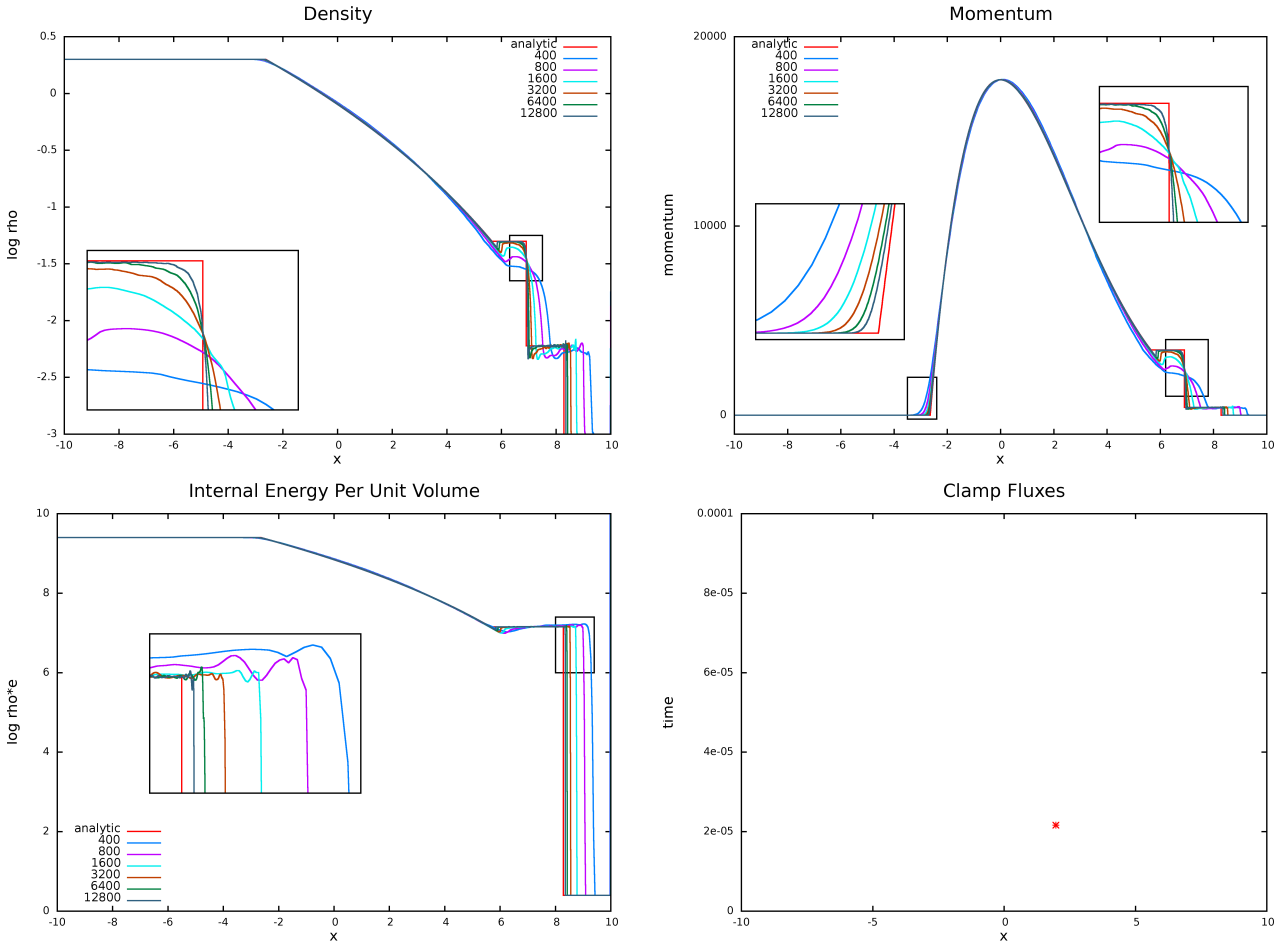


Figure 17: Numerical profiles for the one-dimensional Leblanc shock tube problem at $t = .0001$ using ENO-LLF-3, TVD-RK-2, and CFL=.65 with our adaptive time step restriction. The profiles converge to the analytic solution (shown in red) under grid refinement.

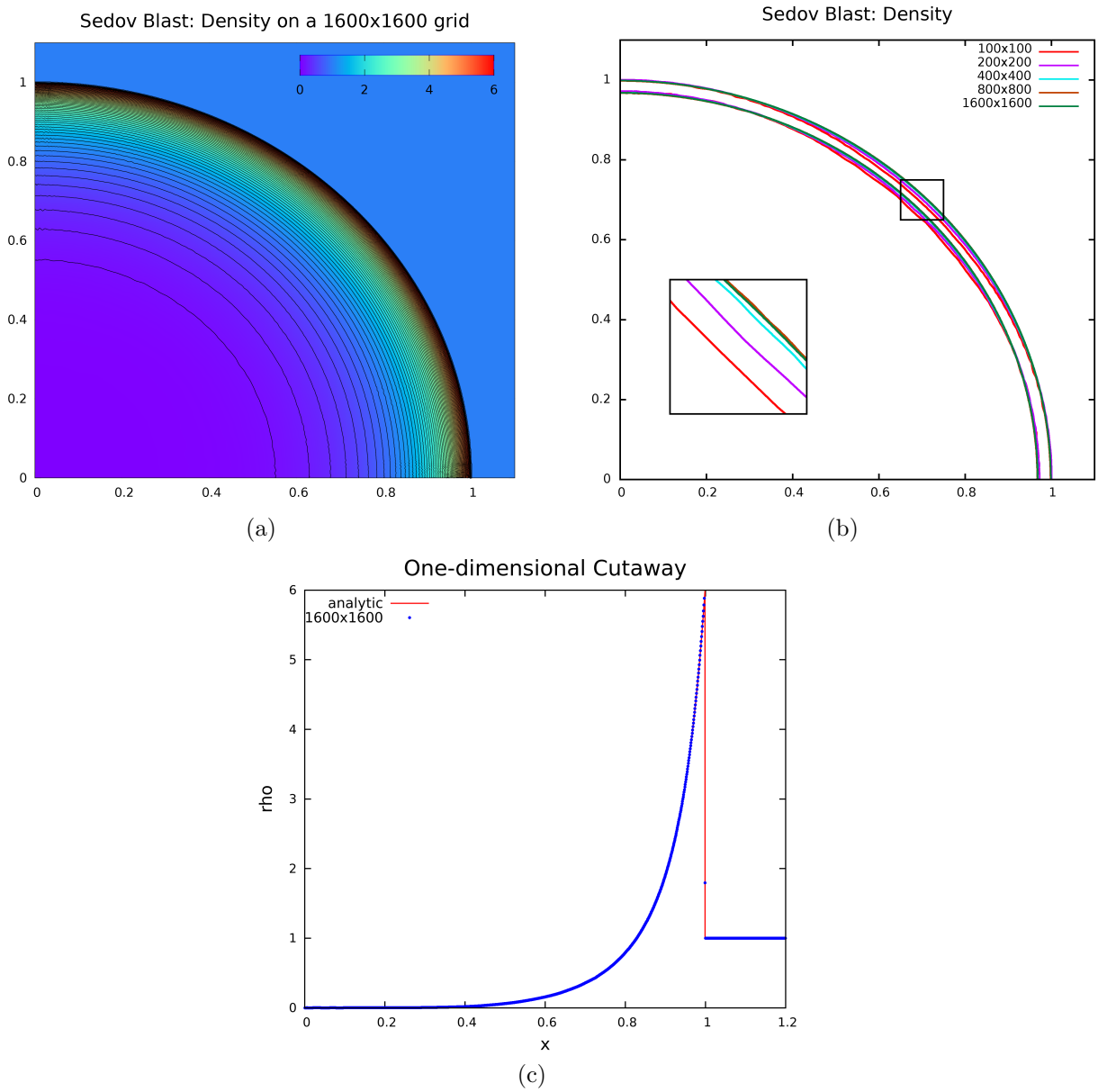


Figure 18: Two-dimensional Sedov blast wave problem. (a) 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 1$. (b) Density contour of $\rho = 4$ at $t = 1$ at various resolutions to illustrate convergence under grid refinement. (c) A one-dimensional cutaway to illustrate the good agreement of our computed numerical solution on a 1600×1600 grid compared to the analytic solution.

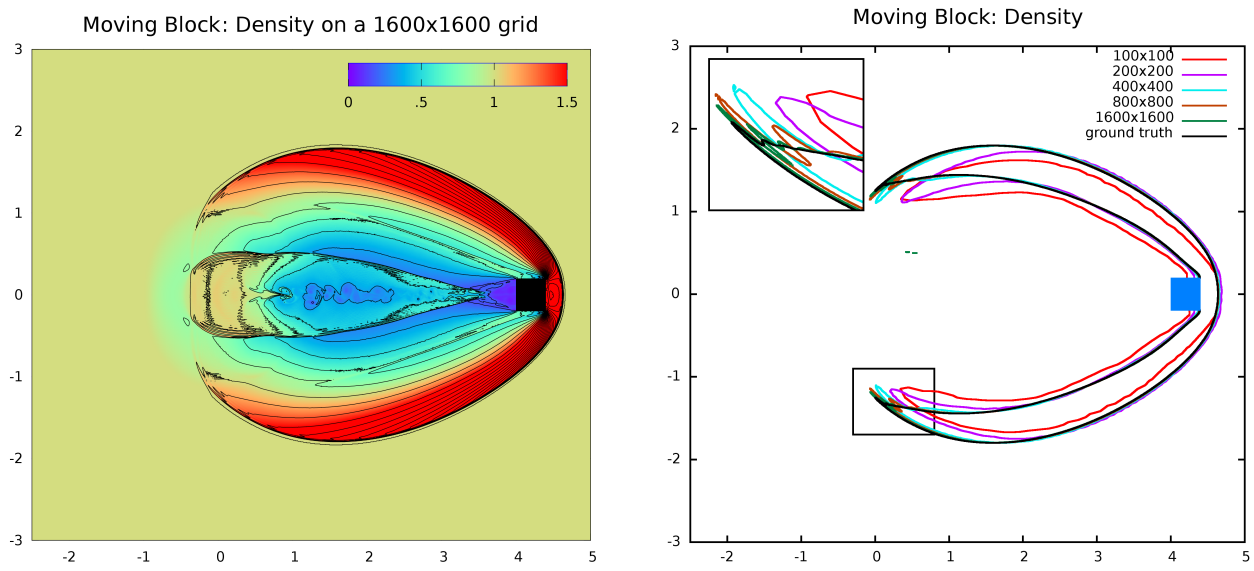


Figure 19: The kinematic block moving in the positive x-direction (left) 30 equally spaced density contours between $\rho = 0$ and $\rho = 5$ at $t = .8$. Note that the color map only goes from $\rho = 0$ to $\rho = 1.5$ to accentuate the details behind the block. (right) Density contour of $\rho = 1.25$ at $t = .8$ at various resolutions to illustrate convergence under grid refinement.

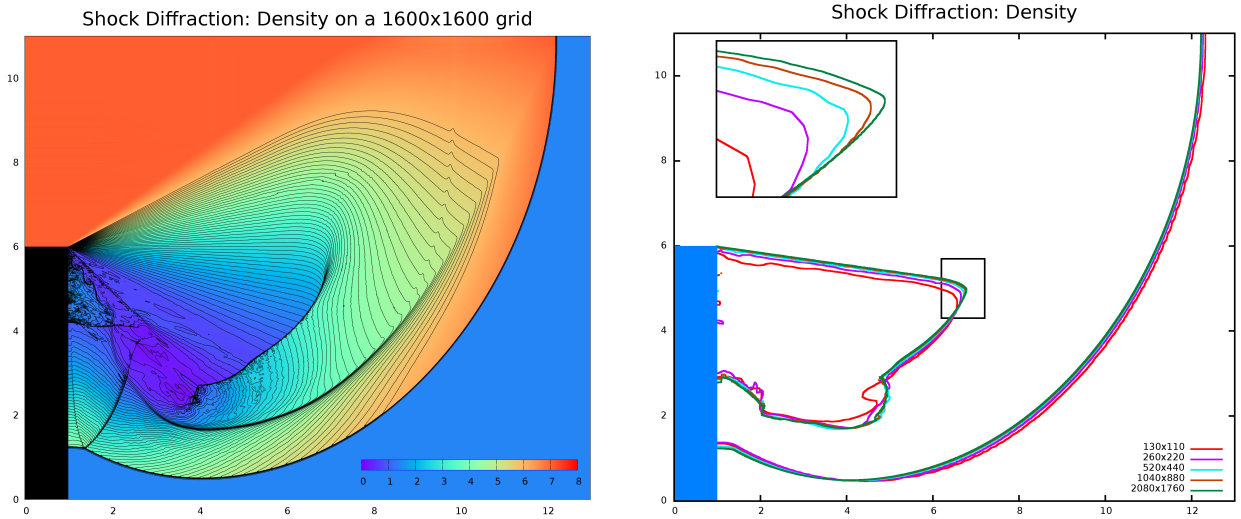


Figure 20: The shock diffraction problem where a shock passes a backward facing corner [6, 41, 44] (left) 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 2.3$. (right) Density contour of $\rho = 1.5$ at $t = 2.3$ at various resolutions to illustrate convergence under grid refinement.

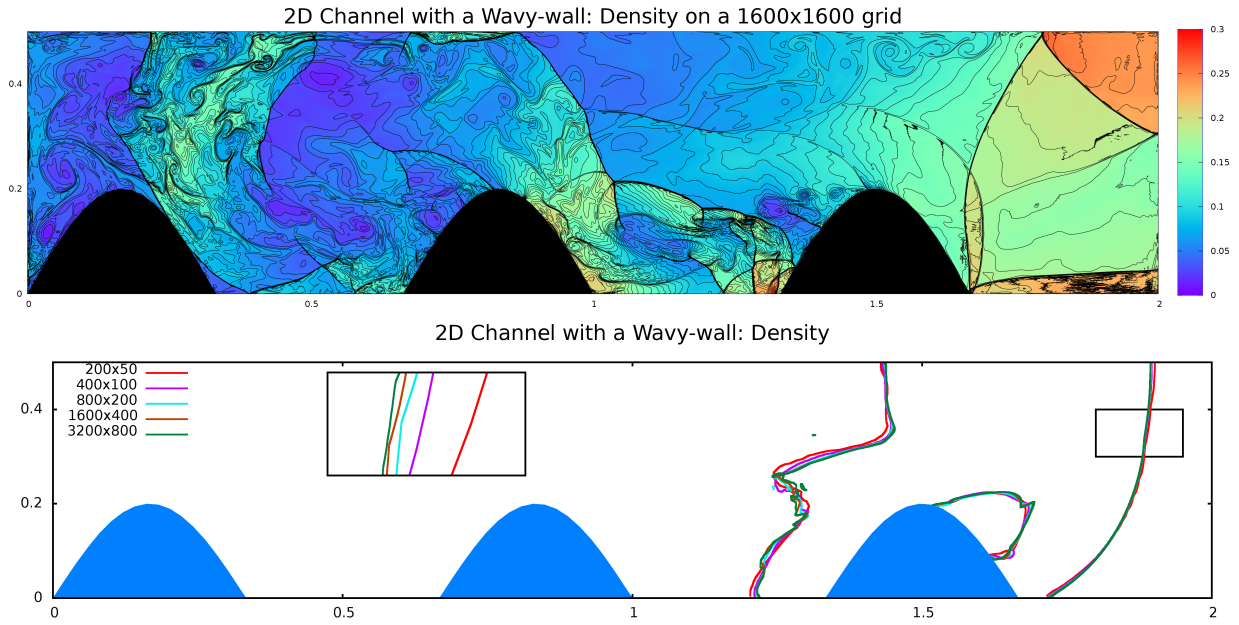


Figure 21: A wavy wall channel similar to [27]. (top) 40 equally spaced density contours between $\rho = 0$ and $\rho = .3$ at $t = .0036$. (bottom) Density contour of $\rho = .16$ at $t = .0011$ at various resolutions to illustrate convergence under grid refinement.

2, TVD-RK-3, and CFL=.5, which is a selection of parameters that allows the code to run to completion without our adaptive time step restriction.

For the shock diffraction problem, Figure 20(left) shows 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = 2.3$ obtained using ENO-LLF-3, TVD-RK-3, and CFL=.5 with our adaptive time step restriction for semi-implicit time integration. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 20(right) shows the density contour of $\rho = 1.5$ at various grid resolutions to illustrate convergence under grid refinement. Note that the results are similar to those in the literature [6, 41, 44]. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-3}$ for resolution 130×110 and successively halved Δt_g each time the resolution was doubled.

For the two dimensional channel with wavy wall problem, Figure 21(top) shows 40 equally spaced density contours between $\rho = 0$ and $\rho = .3$ at $t = .0036$ obtained using ENO-LLF-2, TVD-RK-2, and CFL=.5 with our adaptive time step restriction for semi-implicit time integration. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 21(bottom) shows the density contour of $\rho = .16$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-6}$ for scale 200×50 and successively halved Δt_g each time the resolution was doubled.

For the enclosure problem, Figure 22(top) shows 60 equally spaced density contours between $\rho = 0$ and $\rho = 6$ at $t = .00075$ obtained using ENO-LLF-2, TVD-RK-2, and CFL=.5 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 22(bottom) shows the density contour of $\rho = .05$ at $t = .00075$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 5 \times 10^{-7}$ for resolution 100×50 and successively halved Δt_g each time the resolution was doubled.

8. Two-way solid-fluid coupling

[33] proposed a symmetric positive definite system for handling monolithic two-way solid-fluid coupling for incompressible flow. This method was later extended to compressible flow in [13] by integrating it with

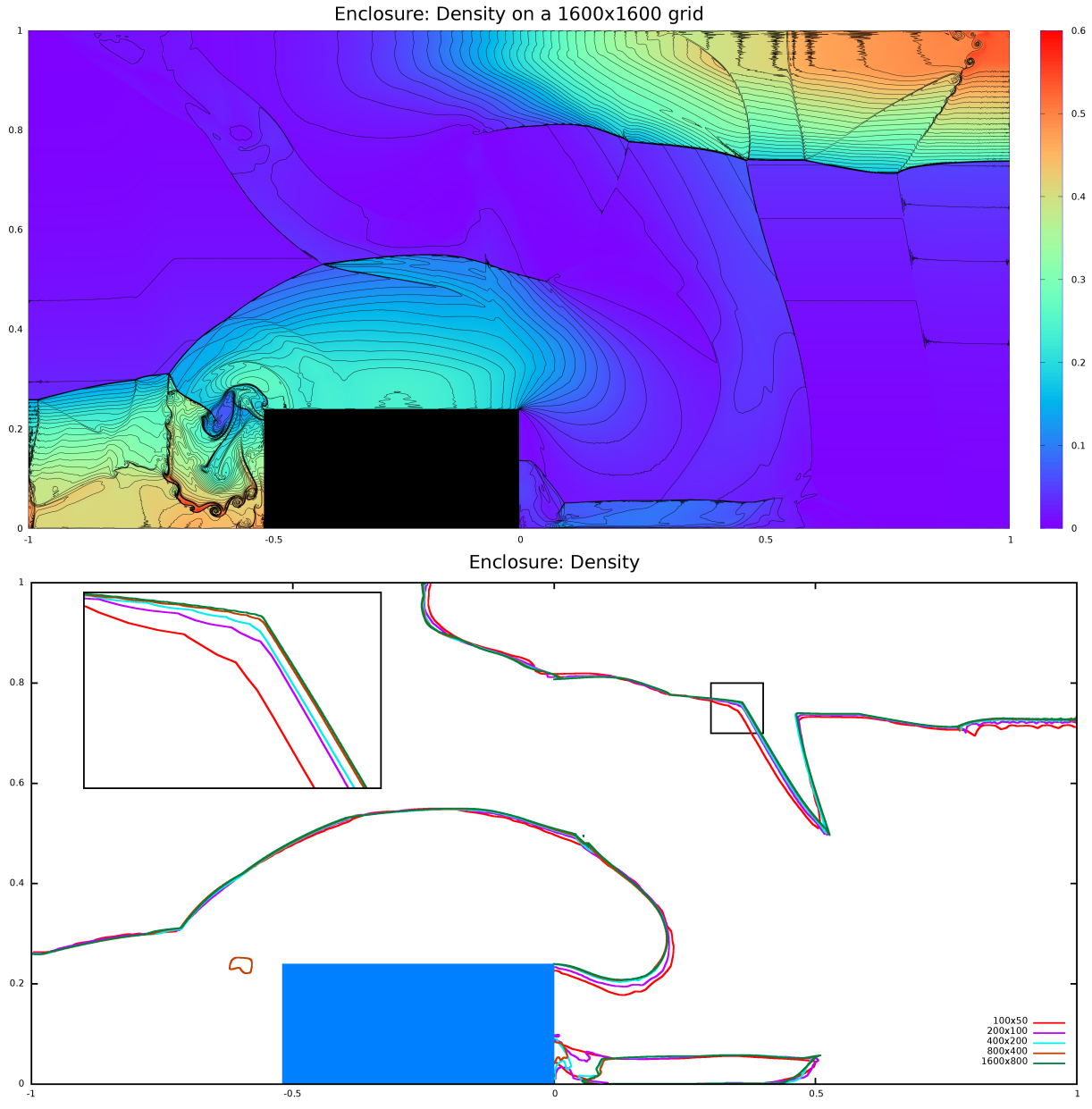


Figure 22: The two dimensional enclosure problem [27]. (top) 60 equally spaced density contours from $\rho = 0$ to $\rho = 6$ at $t = .00075$. (bottom) Density contour of $\rho = .05$ at $t = .00075$ at various grid resolutions to illustrate convergence under grid refinement.

the semi-implicit compressible flow formulation of [22]. The key idea was to use a Lagrange multiplier λ to apply equal and opposite impulses to the solid and fluid. That is, equation (24) is rewritten for all fluid-fluid and solid-fluid dual cells as

$$\vec{u}_f^{n+1} = \vec{u}_f^{\star} - \beta^{-1} G\tilde{p} + \beta^{-1} W^T \lambda, \tag{35}$$

where β is a diagonal matrix of dual cell masses, $-G^T = -V\nabla^T$ is the volume weighted divergence operator defined only for fluid cells, and its negative transpose is the volume weighted gradient G. W is a matrix of 1's and 0's that maps from all non-solid dual cells to all solid-fluid coupled dual cells. Here, solid-fluid coupled dual cells are the partially filled cells on the boundary of the solid where the Lagrange multiplier constraints are applied. [33] incorrectly sets every term in β to be density times the volume of the entire dual cell. To properly address partially filled solid-fluid dual cells, we instead set each term in β to the correct fluid mass in these cells, noting that this is obviously problematic if any term in β goes to zero. To conserve momentum, an equal and opposite impulse $-\lambda$ is applied to the solid,

$$M\vec{v}^{n+1} = M\vec{v}^{\star} - J^T W^T \lambda, \tag{36}$$

while

$$W(J\vec{v}^{n+1} - \vec{u}_f^{n+1}) = 0 \tag{37}$$

constrains the velocity of the solid and the fluid to be equal in solid-fluid dual cells. Here J is an interpolation matrix from solid degrees of freedom to all non-solid dual cells with non-zero rows only for solid-fluid dual cells. Following along the lines of the derivation of equation (29) from equation (24), one may similarly derive

$$[VP^{-1} + G^T\beta^{-1}G]\tilde{p}^{n+1} - G^T\beta^{-1}W^T\lambda = VP^{-1}\tilde{p}^{\star} + G^T\vec{u}_f^{\star} \tag{38}$$

from equation (35). Substituting equations (35) and (36), into equation (37) gives

$$-W\beta^{-1}G\tilde{p}^{n+1} + (W\beta^{-1}W^T + WJM^{-1}J^TW^T)\lambda = WJ\vec{v}^{\star} - W\vec{u}_f^{\star} \tag{39}$$

Finally, equations (38) and (39) can be combined into the following symmetric positive definite system,

$$\begin{pmatrix} VP^{-1} + G^T\beta^{-1}G & -G^T\beta^{-1}W^T \\ -W\beta^{-1}G & W\beta^{-1}W^T + WJM^{-1}J^TW^T \end{pmatrix} \begin{pmatrix} \tilde{p} \\ \lambda \end{pmatrix} = \begin{pmatrix} VP^{-1}\tilde{p}^{\star} + G^T\vec{u}_f^{\star} \\ WJ\vec{v}^{\star} - W\vec{u}_f^{\star} \end{pmatrix} \tag{40}$$

[33] derives

$$J^TG\tilde{p} = M\vec{v}^{\star} + J^T\beta\vec{u}_f^{\star} - (M + J^T\beta J)\vec{v}^{n+1} \tag{41}$$

as an equivalent equation in their system and this plays the same role as the second block equation in the systems proposed in [34] and [14]. The incorrect choice of β in [33] implies that there is too much fluid mass (and thus combined mass) in the solid-fluid region. Unfortunately, using the correct mass drives terms in β to 0 ruining equation (35) which is the first equation in the system in [33]. The systems proposed in [34] and [14] alleviate this by splitting G into G_f (which acts on fluid-fluid faces) and G_s (which acts on solid-fluid coupled faces) so that the vanishing terms in β are associated with G_s and as such do not appear (although [14] ignored the mass of the fluid in solid-fluid coupled dual cells completely). Unfortunately, unlike system (40) which is positive definite, the systems proposed in [34] and [14] are indefinite.

Hence, we propose an approach which remedies the issues with vanishing terms in β. First, we make a change of variables given by $\hat{\lambda} = \lambda + W(\tilde{\beta} - \beta)\vec{u}_f^{n+1} = \lambda + W(\tilde{\beta} - \beta)J\vec{v}^{n+1}$ (using equation (37)), where $\tilde{\beta}$ is a free variable. This allows us to rewrite equation (35) as

$$\vec{u}_f^{n+1} = \hat{\vec{u}}_f^{\star} - \hat{\beta}^{-1}G\tilde{p} + \hat{\beta}^{-1}W^T\hat{\lambda} \tag{42}$$

where $\hat{\beta} = \beta + W^TW(\tilde{\beta} - \beta)$ and $\hat{\vec{u}}_f^{\star} = \hat{\beta}^{-1}\beta\vec{u}_f^{\star}$, and equation (36) as

$$\hat{M}\vec{v}^{n+1} = \hat{M}\hat{\vec{v}}^{\star} - J^TW^T\hat{\lambda} \tag{43}$$

where $\hat{M} = M - J^T W^T W(\tilde{\beta} - \beta)J = M - J^T(\tilde{\beta} - \beta)J$ and $\hat{\vec{v}}^\star = \hat{M}^{-1}M\vec{v}^\star$. Here we used the fact that $W^T W$ is a binary filter that filters out solid-fluid faces from all non-solid faces which is already done by J^T, hence $J^T W^T W = J^T$. Equations (42) and (43) have the same exact forms as equations (35) and (36) respectively and these two sets of equations are exactly equivalent. Hence we can write a system identical to system (40) except with β, $\vec{u}_f^\star$, M, and $\vec{v}^\star$ replaced by $\hat{\beta}$, $\hat{\vec{u}}_f^\star$, $\hat{M}$, and $\hat{\vec{v}}^\star$ respectively. Note that solving this modified system yields the same exact values for p, $\vec{u}_f^{n+1}$, and $\vec{v}^{n+1}$ as would be obtained by solving system (40). Finally, we define the diagonal terms in $\tilde{\beta}$ (a free variable defined above) element-wise via $\tilde{\beta}_i = max(\beta_i, \beta_{min})$ so that the diagonal entries in $\hat{\beta}$ are bounded away from 0 and system (40) can be solved for a β with vanishing terms via a symmetric positive definite system. Note that the terms in $\hat{\beta}$ and $\tilde{\beta}$ corresponding to solid-fluid coupled cells are identical. However, we define $\tilde{\beta}$ in this fashion so that terms corresponding to fluid-fluid dual cells are not clamped if their mass becomes low. Fluid-fluid dual cells with low mass are already properly handled using our aforementioned approach to positivity. Also note that the effective solid mass $\hat{M} = M - J^T(\tilde{\beta} - \beta)J$ closely resembles the lumped mass $M + J^T\beta J$ from [34], except with a negative sign and only non-zero for clamped solid-fluid coupled cells. In fact, clamping $\tilde{\beta}_i = \beta_{min}$ for a given cell i gives a reduction in both the mass and the inertia tensor equivalent to lumping negative mass onto the portion of the solid in cell i in order to compensate for the clamping. Finally, note that $\hat{M}$ is easily symmetric positive definite as long as β_{min} is chosen reasonably small and the rigid body does not have problematic eigenvalues such as would be the case for a long slender rod. (We handle under-resolved bodies in Section 8.1.)

For positivity preservation, the approach from section 7 can be directly applied to two-way solid-fluid coupling since the approach is independent of the actual system being solved for the pressure. Following [31] we set the face pressure to be $\lambda/(dtA)$, where A is the area of a grid face, on solid-fluid coupled faces to compute $\mathbf{F_2(U)}$ instead of using equation (31).

Remark: Note that λ is a Lagrange multiplier as opposed to a pressure, and hence it can attain negative values. Although this is acceptable for the momentum and kinetic energy updates, it can give non-physical answers for internal energy in certain cases. To remedy these cases, the energy in cells bordering the solids can be updated as follows: The energy update for the projection flux is $\nabla \cdot (p\vec{u}) = \vec{u} \cdot \nabla p + p\nabla \cdot \vec{u}$, where the first term updates the kinetic energy and the second term updates potential energy. Noting this, we can store/cache the post advection potential energy before updating the state via $\mathbf{F_2(U)}$. Then the time t^{n+1} kinetic energy can be computed from the time t^{n+1} momentum and density. Next, the cached post advection internal energy is updated to time t^{n+1} using the $p\nabla \cdot \vec{u}$ term where p is clamped to be non-negative. Finally, the time t^{n+1} energy is the sum of the potential and kinetic energy.

Two-dimensional experiment. To test the efficacy of our approach, we simulated the example shown in Figure 23 where two blocks with side length .4 collide with each other. The computational domain is $[-5, 5] \times [-5, 5]$, and initially $\rho = 1$, $u = 0$, and $p = 1$ everywhere. The blocks are placed at $(-2, 0.15)$ and $(2, -0.15)$, have initial velocities of $(5, 0)$ and $(-5, 0)$, and masses of 1. All walls have reflective boundary conditions, and the blocks collide with the bottom wall as well as with each other with a coefficient of restitution of .5. In addition, the blocks are subject to a force field of strength 10 that accelerates them in $-y$ direction. Figure 23 shows 35 equally spaced density contours between $\rho = 0$ and $\rho = 3$ at various times, obtained using ENO-LLF-2, TVD-RK-3, and CFL=.8 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. Figure 24 shows the density contour of $\rho = 2.25$ at $t = .8$ at various resolutions to illustrate convergence under grid refinement. This example required flux clamping, and we used $\Delta t_g = 2 \times 10^{-3}$ for resolution 100×100 and successively halved Δt_g each time the resolution was doubled. Note that we have not used the strategy mentioned in the remark above for running this example.

8.1. Sub-grid Solid-Fluid Coupling

Our main goal is to solve complex solid-fluid coupling problems such as blast waves from explosions impacting complex solid objects. Such explosions are typically characterized by small fragments that fly out with the blast waves and cause spallation (weakening of the material) upon impact. In this section we

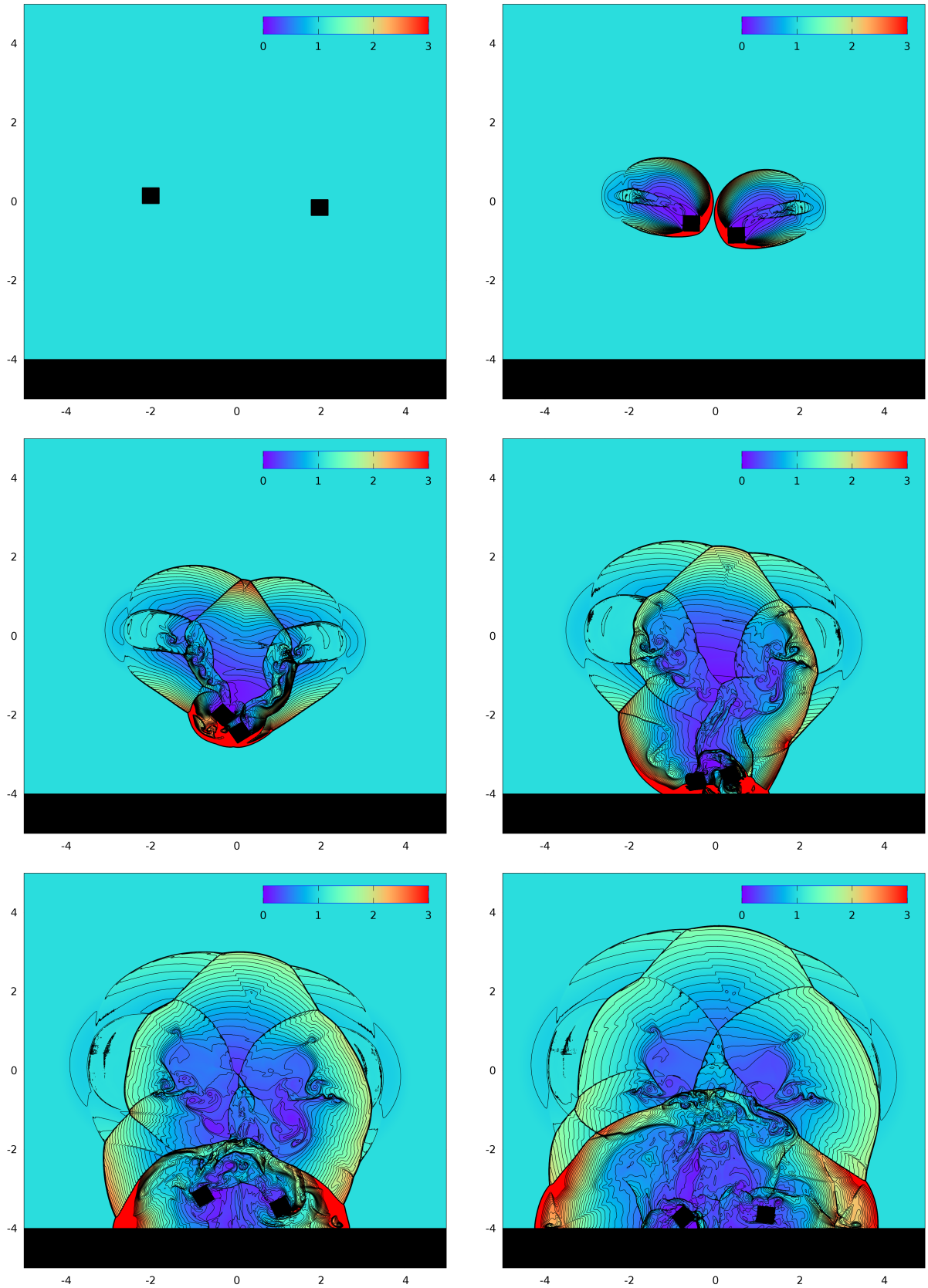


Figure 23: Two blocks collide with each other. 35 equally spaced density contours between $\rho = 0$ and $\rho = 3$ at $t = 0$, $t = .4$, $t = .8$, $t = 1.2$, $t = 1.6$, and $t = 2$ (in row major order).

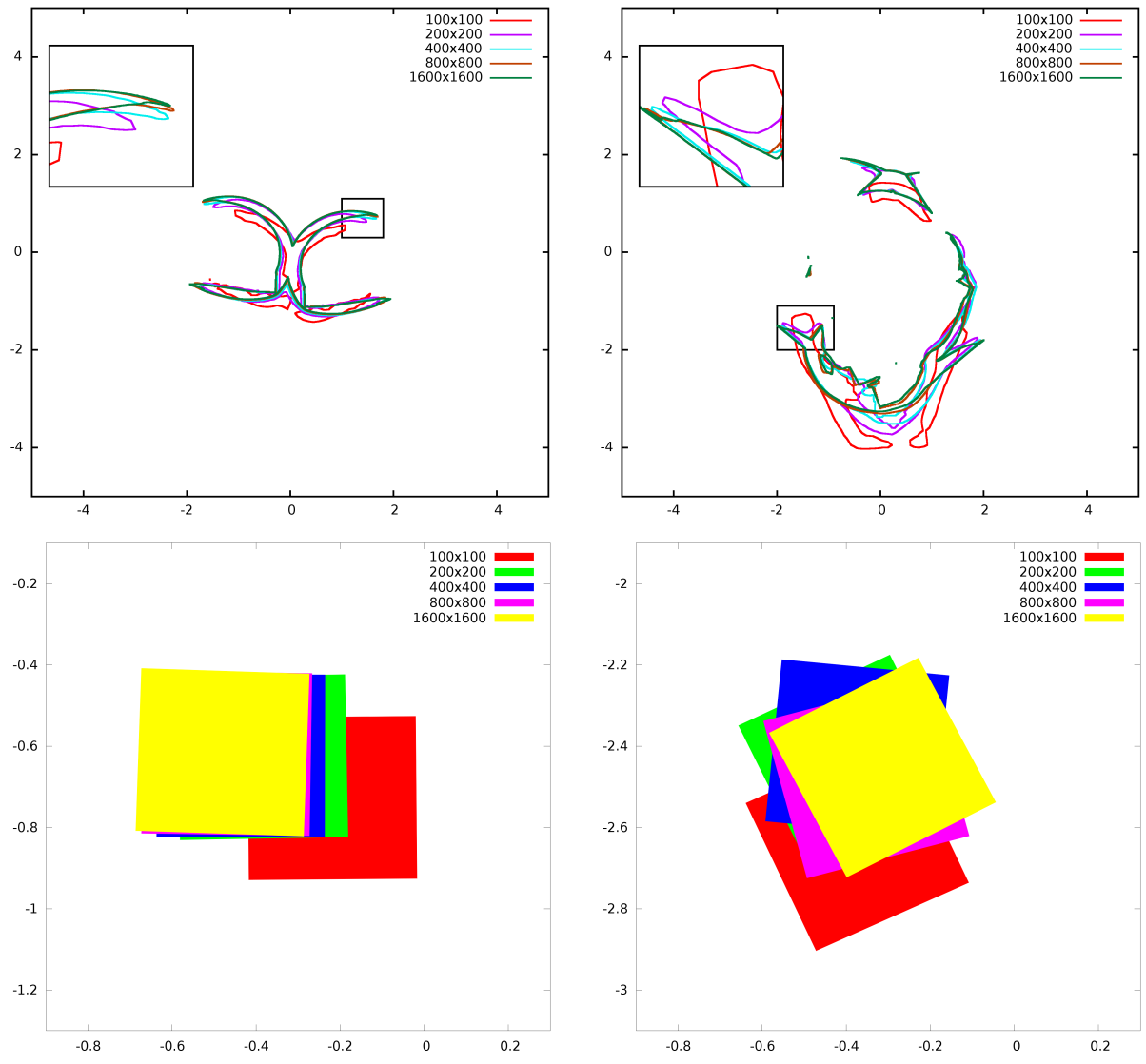
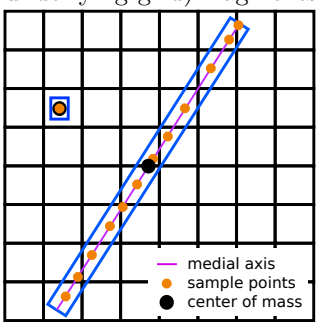


Figure 24: Two blocks collide with each other: (top) Density contour of $\rho = 2.2$ at $t = .42$ (left) and $t = .9$ (right) at various resolutions to illustrate convergence under grid refinement. (bottom) Convergence of the position of the blocks under grid refinement also at $t = .42$ (left) and $t = .9$ (right). Note that the blocks at $t = .9$ do not as readily converge, since the collision not only introduces a discontinuity but also my cause erroneous sticking (see [32]).

explain how our approach handles such small (generally under-resolved by the underlying grid) fragments. This treatment was motivated by the treatment of small bubbles in [28].



We model the sub-grid solid-fluid coupling by writing the velocity matching equation at sample points on solids rather than at dual cell centers (as was done in equation (37)). For small solids we use the center of mass as the sample point, and for thin rod-like structures we use one sample per cell placed at the center of the portion of its medial axis that lies in that cell (see figure on the right). Let J_s be the interpolation operator from the solid degrees of freedom to the sample points. We then define H to be the interpolation operator from fluid faces to the same sample points. An equal and opposite impulse is applied to each sample point and mapped back to the solid and fluid via J_s^T and H^T respectively, i.e.,

$$\vec{u}_f^{n+1} = \vec{u}_f^{\star} - \beta^{-1} G\tilde{p} + \beta^{-1} H^T \lambda_s, \tag{44}$$

$$M_s \vec{v}_s^{n+1} = M_s \vec{v}_s^{\star} - J_s^T \lambda_s. \tag{45}$$

$$J_s \vec{v}^{n+1} - H\vec{u}_f^{n+1} = 0. \tag{46}$$

where equation (46) is written only for our chosen sample points in contrast to equation (37). However, if one draws an equivalence between H and W as well as between J_s and WJ, one can see the similarities with the prior approach. Similar to system (40), this gives a symmetric positive definite system. Combining the equations for sub-grid rigid bodies with system (40) for well resolved rigid bodies, yields

$$\begin{pmatrix} VP^{-1} + G^T\hat{\beta}^{-1}G & -G^T\hat{\beta}^{-1}W^T & -G^T\hat{\beta}^{-1}H^T \\ -W\hat{\beta}^{-1}G & W\hat{\beta}^{-1}W^T + WJ\hat{M}^{-1}J^TW^T & 0 \\ -H\hat{\beta}^{-1}G & 0 & H\hat{\beta}^{-1}H^T + J_s(\alpha M_s)^{-1}J_s^T \end{pmatrix} \begin{pmatrix} \tilde{p} \\ \hat{\lambda} \\ \lambda_s \end{pmatrix} = \begin{pmatrix} VP^{-1}\tilde{p}^{\star} + G^T\hat{\vec{u}}_f^{\star} \\ WJ\hat{\vec{v}}^{\star} - W\hat{\vec{u}}_f^{\star} \\ J_s\vec{v}_s^{\star} - H\hat{\vec{u}}_f^{\star} \end{pmatrix} \tag{47}$$

after replacing appropriate variables with hats in order to properly handle partially fluid-filled dual cells. This formulation works well in practice except that small solids at high velocities may be slowed down too much by the large amount of fluid in the surrounding grid cell. Although this would be alleviated to some degree by grid refinement, we propose the following modification to equation (46) on coarse grids,

$$J_s\vec{v}_s^{n+1} - \alpha H\vec{u}_f^{n+1} - (1-\alpha)J_s\vec{v}_s^{\star} = 0. \tag{48}$$

One can show that this is mathematically equivalent to dividing a solid of mass M_s into two pieces with masses αM_s and $(1-\alpha)M_s$, where the first piece two-way couples with the fluid equilibrating its velocity and the second piece continues traveling with its time $t^{\star}$ velocity; afterwards, the two pieces are combined via an inelastic collision. Equation (48) is the source of the α factor in system (47).

After solving system (47), we update the face velocities using equation (44) and ignore the contribution of λ_s to the pressure when constructing $\mathbf{F}_2(\mathbf{U})$. Then, we update the fluid to time t^{n+1}, and store/cache the time t^{n+1} potential energy. To account for the effect of λ_s on the momentum and energy, we compute $H^T\lambda_s$ to find the impulse which is applied to fluid faces and distribute this face impulse from every face to the two cells bordering that face in a density-weighted manner. Applying this cell impulse changes the momentum, and we use this new momentum to compute the kinetic energy. The total energy in each cell is then the sum of this kinetic energy and the cached potential energy. Algorithm 1 can now be modified as shown in algorithm 2 to handle solid-fluid coupling and sub-grid bodies.

Two-dimensional experiments. We show two examples with sub-grid rigid bodies demonstrating that our method can be used to maintain positivity even in the presence of these sub-grid bodies. Consider a domain of $[0,1]\times[0,.5]$ as shown in Figure 8.1 (right). Initial conditions are $p = 10^4$, $\rho = .1$, and $v = 0$ everywhere. If $x \leq .3$, then $u = 100$, otherwise $u = 0$. We place 200 heavy (mass .2) followed by 100 light (mass 2×10^{-5}) followed by 200 heavy (mass .2) sub-grid rigid bodies all having side length 5×10^{-4} at the center of the domain as shown in Figure 8.1 (left). In the gap between the light bodies we place 2 sub-grid rods of mass

Algorithm 2 Final Simulation Loop

while $time < t_{target}$ **do**
  Compute the time step size Δt.
  $U_{save} = U^n$.
  Update U^n with a positivity preserving forward Euler step of size Δt_1 and fluxes $\mathbf{F}_{11}$.
  Take another positivity preserving forward Euler step of size Δt_2 and fluxes $\mathbf{F}_{12}$.
  Compute the effective advection flux $\mathbf{F}_1(\mathbf{U})$ as described in equation (34).
  Compute $\Delta t_{\text{adv}} = \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}$.
  Solve the system in equation (47) with $\Delta t = \Delta t_{\text{adv}}$ to obtain p, $\hat{\lambda}$, and λ_s.
  Update fluid face velocities $\vec{u}_f$ to time t^{n+1} using equations (35) and (44).
  Compute p_f as per equation (31), and use $p_f = \lambda/(dtA)$ on solid-fluid faces.
  Use p_f along with the updated face velocities to construct the projection flux $\mathbf{F}_2(\mathbf{U}) = (0, p_f, p_f \vec{u}_f)^T$.
  Compute the effective flux $\mathbf{F}_{\text{eff}} = \mathbf{F}_1(\mathbf{U}) + \mathbf{F}_2(\mathbf{U})$.
  Use adaptive time step restriction on $U = U_{save}$ with fluxes $\mathbf{F}_{\text{eff}}$ to obtain Δt_{final}.
  if $\Delta t_{\text{final}} < \Delta t_g$ **then**
    $\Delta t_{\text{final}} = \Delta t_g$
    Perform flux clamping on $\mathbf{F}_{\text{eff}}$ to obtain $\hat{\mathbf{F}}_{\text{eff}}$
  end if
  Update U_{save} to U^{n+1} with time step Δt_{final} and fluxes $\hat{\mathbf{F}}_{\text{eff}}$ similar to equation (33)
  Store/cache the potential energy E_p.
  Compute face impulse $I^f = H^T \lambda_s$.
  For every face $i+1/2$ distribute $I^f_{i+1/2}$ to two neighboring cells, $I_i \mathrel{+}= \frac{\rho_i I^f_{i+1/2}}{\rho_i + \rho_{i+1}}$ and $I_{i+1} \mathrel{+}= \frac{\rho_{i+1} I^f_{i+1/2}}{\rho_i + \rho_{i+1}}$.
  Apply the cell impulse I to the momentum in each cell, use the updated momentum to compute the kinetic energy E_k.
  Compute the total energy in cell as $E_p + E_k$.
  Use the old ρ along with the updated momentum and energy to construct the final time t^{n+1} state.
  Compute the final solid velocities $\vec{v}^{n+1}$ as per equations (36) and (45).
  Advance the rigid bodies with contacts and collisions as per [15].
  $time \mathrel{+}= \Delta t_{\text{final}}$.
end while

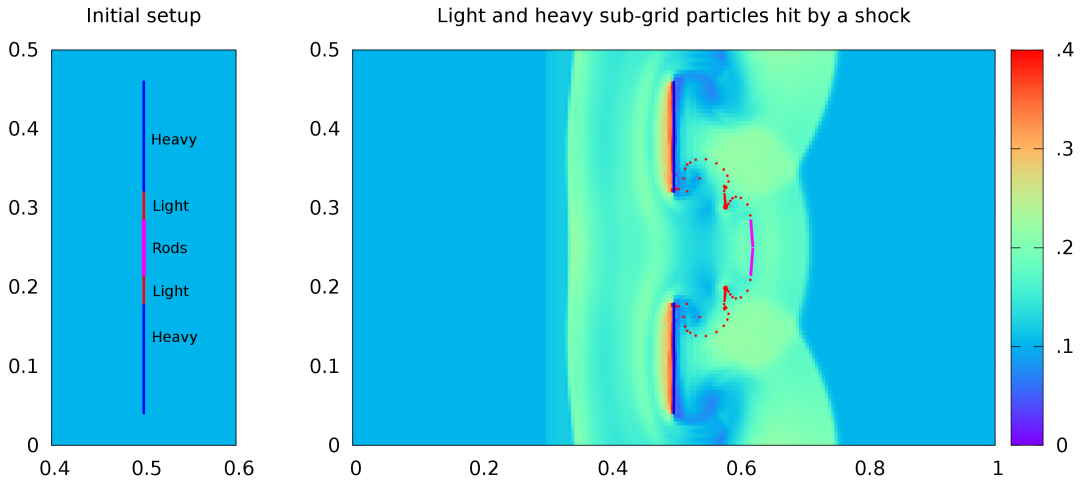


Figure 25: (left) Initial setup. (right) Density contours between $\rho = 0$ and $\rho = .4$ for 400 heavy (blue) and 100 light (red) "small" sub-grid bodies, and 2 "rod-like" sub-grid bodies (magenta) hit by a high velocity fluid in two spatial dimensions on a 200×100 grid at time $t = .008$.

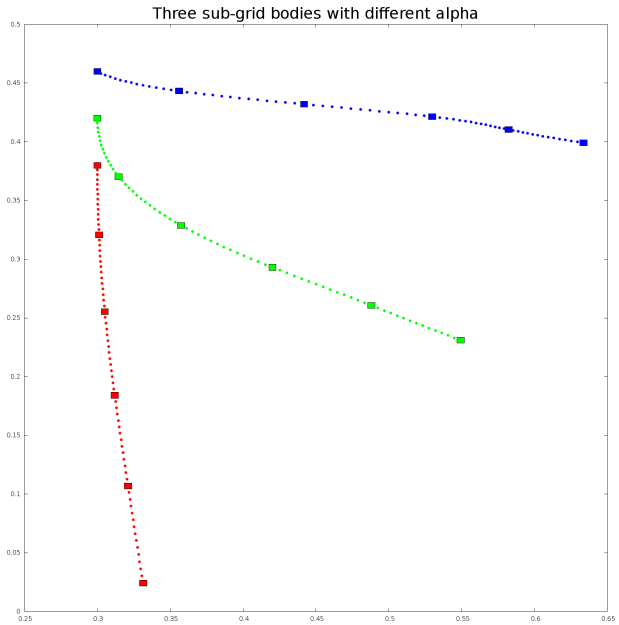


Figure 26: Time evolution from $t = 0$ to $t = 0.14$ of three sub-grid rigid bodies with different values of α hit by a high velocity fluid in two spatial dimensions on a 100×25 grid.

5×10^{-4} and dimensions $(5 \times 10^{-4}) \times .035$. The boundary conditions are reflective everywhere. Figure 8.1 (right) shows the density color map at $t = .0008$ obtained using ENO-LLF-2, TVD-RK-3, and CFL=.5 with our adaptive time step restriction. This choice of scheme and parameters is unable to run to completion without our adaptive time step restriction. This example required flux clamping, and we used $\Delta t_g = 1 \times 10^{-6}$ for scale 200×100. Next, consider a domain of $[0, 2] \times [0, .5]$ as shown in Figure 8.1. Initial conditions are $p = 10^2$, $\rho = .1$, and $v = 0$ everywhere. If $x \leq .3$, then $u = 5$, otherwise $u = 0$. We place three sub-grid rigid bodies all with side length 5×10^{-3} and mass 2×10^{-4} at $x = .3$ and $y = .38$, $y = .42$, and $y = .46$ respectively. The bodies have $\alpha = .001$, $\alpha = .015$, and $\alpha = 1.$ respectively. Figure 8.1 shows the time evolution from $t = 0$ to $t = 0.14$ of these bodies on a 100×25 grid to demonstrate the effect of varying α.

9. Conclusion

We designed a novel method which adaptively clamps the size of the time step in order to guarantee positive density and internal energy. To prevent the time step size from becoming arbitrarily small, we also deigned a local conservative flux clamping scheme. We demonstrated the usefulness of our method on several one-dimensional and two-dimensional problems. Since our method takes the form of a time step restriction, it is applicable to any spatial scheme using a method of lines approach. It can also be used with any equation of state.

Acknowledgements

Research supported in part by ONR N00014-13-1-0346, ONR N00014-11-1-0707, ONR N-00014-11-1-0027, ONR N00014-09-1-0101, ARL AHPCRC W911NF-07-0027, and the Intel Science and Technology Center for Visual Computing. M. A. was supported in part by the Nokia Research Center. Computing resources were provided in part by ONR N00014-05-1-0479. We would like to thank Prof. Xiangxiong Zhang and Aamer Haque for providing us with the analytic solutions to the Sedov blast wave problem.

10. Bibliography

[1] Christophe Berthon. Stability of the MUSCL schemes for the Euler equations. *Communications in Mathematical Sciences*, 3:133–157, 2005.

[2] Christophe Berthon. Robustness of MUSCL schemes for 2D unstructured meshes. *J. Comput. Phys.*, 218:495–509, 2006.

[3] Bram and van Leer. Towards the ultimate conservative difference scheme V. A second-order sequel to Godunov's method. *J. Comput. Phys.*, 135:229 – 248, 1997.

[4] Juan Cheng and Chi-Wang Shu. Positivity-preserving lagrangian scheme for multi-material compressible flow. *J. Comput. Phys.*, 257, Part A(0):143 – 168, 2014.

[5] B. Cockburn, S.-Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems. *J. Comput. Phys.*, 84:90–113, 1989.

[6] Bernardo Cockburn and Chi-Wang Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V multidimensional systems. *J. Comput. Phys.*, 141:199–224, 1998.

[7] Bernd Einfeld. On Godunov-type methods for gas dynamics. *SIAM J. Numer. Anal.*, 25:294–318, 1988.

[8] B. Einfeldt, C. D. Munz, P. L. Roe, and B. Sjögreen. On Godunov-type methods near low densities. *J. Comput. Phys.*, 92:273–295, 1991.

[9] R. Fedkiw, X.-D. Liu, and S. Osher. A general technique for eliminating spurious oscillations in conservative schemes for multiphase and multispecies Euler equations. *Int. J. Nonlinear Sci. and Numer. Sim.*, 3:99–106, 2002.

[10] R. Fedkiw, B. Merriman, and S. Osher. Numerical methods for a mixture of thermally perfect and/or calorically perfect gaseous species with chemical reactions. *J. Comput. Phys.*, 132:175–190, 1997.

[11] F. Gibou and C. Min. Efficient symmetric positive definite second-order accurate monolithic solver for fluid/solid interactions. *J. Comput. Phys.*, 231(8):3246–3263, 2012.

[12] Jrmie Gressier, Philippe Villedieu, and Jean-Marc Moschetta. Positivity of flux vector splitting schemes. *J. Comput. Phys.*, 155(1):199 – 220, 1999.

[13] J. Grétarsson and R. Fedkiw. Fully conservative, robust treatment of thin shell fluid-structure interactions in compressible flows. *J. Comput. Phys.*, 245:160–204, 2013.

[14] J.T. Grétarsson, N. Kwatra, and R. Fedkiw. Numerically stable fluid-structure interactions between compressible flow and solid structures. *J. Comput. Phys.*, 230:3062–3084, 2011.

[15] E. Guendelman, R. Bridson, and R. Fedkiw. Nonconvex rigid bodies with stacking. *ACM TOG*, 22(3):871–878, 2003.

[16] A. Harten, B. Enquist, S. Osher, and S. Chakravarthy. Uniformly high-order accurate essentially non-oscillatory schemes III. *J. Comput. Phys.*, 71:231–303, 1987.

[17] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov type schemes for hyperbolic conservation laws. *SIAM Review*, 25:35–61, 1983.

[18] Michael T. Heath. *Scientific Computing*. Mc-Graw Hill, USA, 2002.

[19] Xiangyu Y. Hu, Nikolaus A. Adams, and Chi-Wang Shu. Positivity-preserving method for high-order conservative schemes solving compressible euler equations. *J. Comput. Phys.*, 242(0):169 – 180, 2013.

[20] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.

[21] V. P. Korobeinikov. *Problems of Point-Blast Theory.* Springer, 2007.

[22] N. Kwatra, J. Su, J.T. Grétarsson, and R. Fedkiw. A method for avoiding the acoustic time step restriction in compressible flow. *J. Comput. Phys.*, 228(11):4146–4161, 2009.

[23] T. Linde and P. L. Roe. Robust Euler codes. In *13th Computational Fluid Dynamics Conference*, pages 83–93. AIAA, 1997.

[24] Wei Liu, Juan Cheng, and Chi-Wang Shu. High order conservative lagrangian schemes with lax-wendroff type time discretization for the compressible Euler equations. *J. Comput. Phys.*, 228(23):8872–8891, 2009.

[25] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 126:202–212, 1996.

[26] Bernard Parent. Positivity-preserving flux-limited method for compressible fluid flow. *Computers and Fluids*, 44(1):238 – 247, 2011.

[27] Bernard Parent. Positivity-preserving high-resolution schemes for systems of conservation laws. *J. Comput. Phys.*, 231(1):173 – 189, 2012.

[28] Saket Patkar, Mridul Aanjaneya, Dmitriy Karpman, and Ronald Fedkiw. A hybrid Lagrangian-Eulerian formation for bubble generation and dynamics. In *Proc. of the 2013 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, pages 105–114, 2013.

[29] B. Perthame. Second-order Boltzmann schemes for compressible Euler equations in one and two space dimensions. *SIAM Journal on Numerical Analysis*, 29:1–19, 1992.

[30] B. Perthame and C.-W. Shu. On positivity preserving finite volume schemes for Euler equations. *Numerische Mathematik*, 73:119–130, 1996.

[31] L. Qiu, W. Lu, and R. Fedkiw. An adaptive discretization of compressible flow using a multitude of moving Cartesian grids. *(submitted)*, 2014.

[32] L. Qiu, Y. Yu, and R. Fedkiw. On thin gaps between rigid bodies two-way coupled to incompressible flow. *J. Comput. Phys.*, 292(0):1 – 29, 2015.

[33] A. Robinson-Mosher, C. Schroeder, and R. Fedkiw. A symmetric positive definite formulation for monolithic fluid structure interaction. *J. Comput. Phys.*, 230(4):1547–1566, 2011.

[34] A. Robinson-Mosher, T. Shinar, J. T. Grétarsson, J. Su, and R. Fedkiw. Two-way coupling of fluids to rigid and deformable solids and shells. *ACM Trans. Graph. (SIGGRAPH Proc.)*, 27(3):46:1–46:9, August 2008.

[35] R. Sanders. A third-order accurate variation nonexpansive difference scheme for single nonlinear conservation law. *Mathematics of Computation*, 51:535–558, 1988.

[36] L. I. Sedov. *Similarity and Dimensional Methods in Mechanics.* CRC Press, 1993.

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

[38] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes II (two). *J. Comput. Phys.*, 83:32–78, 1989.

[39] Xiangxiong Zhang and Chi-Wang Shu. A genuinely high order total variation diminishing scheme for one-dimensional scalar conservation laws. *SIAM Journal on Numerical Analysis*, 48:772–795, 2010.

[40] Xiangxiong Zhang and Chi-Wang Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *J. Comput. Phys.*, 229:3091–3120, 2010.

[41] Xiangxiong Zhang and Chi-Wang Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *J. Comput. Phys.*, 229:8918–8934, 2010.

[42] Xiangxiong Zhang and Chi-Wang Shu. Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 467(2134):2752–2776, 2011.

[43] Xiangxiong Zhang and Chi-Wang Shu. Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms. *J. Comput. Phys.*, 230:1238–1248, 2011.

[44] Xiangxiong Zhang and Chi-Wang Shu. Positivity-preserving high order finite difference WENO schemes for compressible euler equations. *J. Comput. Phys.*, 231(5):2245 – 2258, 2012.

[45] Xiangxiong Zhang, Yinhua Xia, and Chi-Wang Shu. Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50(1):29–62, 2012.

Appendix I

Consider equation (16). Rewriting L_1 and L_2 in terms of a continuous function L gives

$$\phi^{n+1} = \phi^n + \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}\left(\frac{\Delta t_1 L(t,\phi^n) + \Delta t_2 L(t+\Delta t_1, \phi^n + \Delta t_1 L(t,\phi^n))}{\Delta t_1 + \Delta t_2}\right).$$

Taylor expanding $L(t+\Delta t_1, \phi^n + \Delta t_1 L(t,\phi^n))$ around $L(t,\phi^n)$ as,

$$L(t+\Delta t_1, \phi^n + \Delta t_1 L(t,\phi^n)) = L(t,\phi^n) + \Delta t_1 \frac{\partial L}{\partial t} + \Delta t_1 L(t,\phi^n)\frac{\partial L}{\partial \phi} + \mathcal{O}(\Delta t_1^2)$$

and substituting the result in the expression for ϕ^{n+1} gives

$$\phi^{n+1} = \phi^n + \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2} L(t,\phi^n) + \frac{1}{2}\left(\frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}\right)^2 \left(\frac{\partial L}{\partial t} + L(t,\phi^n)\frac{\partial L}{\partial \phi}\right) + \frac{1}{2\Delta t_1}\left(\frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}\right)^2 \mathcal{O}(\Delta t_1^2).$$

Substituting $\Delta t = \frac{2\Delta t_1 \Delta t_2}{\Delta t_1 + \Delta t_2}$ and using the fact that $\Delta t \leq \Delta t_1$ gives

$$\phi^{n+1} = \phi^n + \Delta t L(t,\phi^n) + \frac{1}{2}\Delta t^2 \left(\frac{\partial L}{\partial t} + L(t,\phi^n)\frac{\partial L}{\partial \phi}\right) + \mathcal{O}(\Delta t^3)$$

which is precisely the Taylor expansion for ϕ^{n+1} accurate to second order.

Appendix II

In [37] the authors write all possible RK-3 updates parametrized as follows

$$
\begin{aligned}
u^{(0)} &= u^n && (49)\\
u^{(1)} &= u^{(0)} + \Delta t \beta_{10} L(u^{(0)}) && (50)\\
u^{(2)} &= u^{(0)} + \Delta t((\beta_{20} + \alpha_{21}\beta_{10})L(u^{(0)}) + \beta_{21} L(u^{(1)})) && (51)\\
u^{(3)} &= u^{(0)} + \Delta t((\beta_{30} + \alpha_{31}\beta_{10} + \alpha_{32}(\beta_{20} + \alpha_{21}\beta_{10}))L(u^{(0)}) + (\beta_{31} + \alpha_{32}\beta_{21})L(u^{(1)}) + \beta_{32} L(u^{(2)})) && (52)
\end{aligned}
$$

Without loss of generality we can define $P = \beta_{20} + \alpha_{21}\beta_{10} + \beta_{21}$ as in [37]. One can also derive the following condition by eliminating β_{32} from the first two equations in equation 2.17 of [37]

$$P = \beta_{10} + 2\beta_{10}\frac{\beta_{21}}{P} - 3\beta_{10}^2\frac{\beta_{21}}{P}. \tag{53}$$

In Section 4, we also make use of the auxiliary variables u^2, u^3, Δt_1, Δt_2, and Δt_3 via

$$
\begin{aligned}
u^{(1)} &= u^{(0)} + \Delta t_1 L(u^{(0)}) && (54)\\
u^2 &= u^{(1)} + \Delta t_2 L(u^{(1)}) = u^{(1)} + k\Delta t_1 L(u^{(1)}) && (55)\\
u^3 &= u^{(2)} + \Delta t_3 L(u^{(2)}) && (56)
\end{aligned}
$$

where $\Delta t_1 = \Delta t \beta_{10}$ from equation (50), and $k = \Delta t_2/\Delta t_1 \geq 0$ is determined after Δt_1 and Δt_2 are chosen using our adaptive time step restriction described in Section 3. Note that our adaptive time step restriction guarantees the positivity of $u^{(1)}$, u^2, and u^3. Given this information, our goal is to obtain positive values for $u^{(2)}$ and $u^{(3)}$.

Equation (51) can be rewritten as,

$$u^{(2)} = u^{(0)} + \Delta t((P - \beta_{21})L(u^{(0)}) + \beta_{21} L(u^{(1)})) \tag{57}$$

Using equations (54) and (55) to eliminate $L(u^{(0)})$ and $L(u^{(1)})$ we can rewrite this as

$$u^{(2)} = \left(1 - \frac{\Delta t(P - \beta_{21})}{\Delta t_1}\right) u^{(0)} + \left(\frac{\Delta t(P - \beta_{21})}{\Delta t_1} - \frac{\Delta t \beta_{21}}{\Delta t_2}\right) u^{(1)} + \frac{\Delta t \beta_{21}}{\Delta t_2} u^2 \tag{58}$$

Positivity of the coefficient[3] of u^2 implies $\beta_{21} \geq 0$, while positivity of the coefficient of $u^{(1)}$ implies $(P - \beta_{21})\Delta t_2 \geq \beta_{21}\Delta t_1$ or $(P - \beta_{21})\Delta t_2 = n\beta_{21}\Delta t_1$, for some $n \geq 1$. Thus $\beta_{21} = P\Delta t_2/(n\Delta t_1 + \Delta t_2) = Pk/(n+k)$ allowing us to rewrite equation (57) as

$$u^{(2)} = u^{(0)} + \Delta t P\left(\frac{n\Delta t_1 L(u^{(0)}) + \Delta t_2 L(u^{(1)})}{n\Delta t_1 + \Delta t_2}\right) \tag{59}$$

Since the update for $u^{(2)}$ in RK-3 is similar to that for $u^{(2)}$ in RK-2, comparing equation (59) with equations (15), (16), and (17) motivates the reparametrization of $P\Delta t$ as

$$P\Delta t = \frac{m\Delta t_1 \Delta t_2}{n\Delta t_1 + \Delta t_2} = \frac{m\Delta t_1 k}{n + k} \tag{60}$$

in terms of a new parameter m. Substituting β_{10} and β_{21} into equation (53) results in

$$P = \frac{(n\Delta t + 3k\Delta t - 3k\Delta t_1)\Delta t_1}{\Delta t^2(n + k)} \tag{61}$$

[3]This is a sufficient but not a necessary condition for positivity preservation. However, if the coefficients are not all positive, then they would have to depend on the state, which is not very general.

Equations (60) and (61) together allow us to find P and Δt as

$$P = \frac{(3k + n - km)m}{3(n + k)} \tag{62}$$

$$\Delta t = \frac{3k\Delta t_1}{3k + n - km} \tag{63}$$

Finally, positivity of the coefficient of $u^{(0)}$ implies $\Delta t(P - \beta_{21}) \leq \Delta t_1$ or $\Delta t P n/(n + k) \leq \Delta t_1$. Thus from equation (60), $mkn \leq (n + k)^2$ which can be rewritten as $n^2 + (2 - m)kn + k^2 \geq 0$. When $m = 4$, one can complete the square guaranteeing the satisfaction of the above condition for all n and k. Thus, choosing $m \leq 4$ guarantees the positivity of the coefficient of $u^{(0)}$.

Next, we proceed applying similar rules to $u^{(3)}$. Letting $k_1 = \Delta t(P - \beta_{21})$, $k_2 = \Delta t\beta_{21}$, $k_3 = \Delta t(\beta_{30} + \alpha_{31}\beta_{10} + \alpha_{32}(\beta_{20} + \alpha_{21}\beta_{10}))$, $k_4 = \Delta t(\beta_{31} + \alpha_{32}\beta_{21})$, and $k_5 = \Delta t\beta_{32}$ gives

$$\begin{aligned} u^{(3)} &= u^{(0)} + k_3 L(u^{(0)}) + k_4 L(u^{(1)}) + k_5 L(u^{(2)}) && (64)\\ &= u^{(0)} + k_3\frac{(u^{(1)} - u^{(0)})}{\Delta t_1} + k_4\frac{(u^2 - u^{(1)})}{\Delta t_2} + k_5\frac{(u^3 - u^{(2)})}{\Delta t_3} && (65)\\ &= (1 - \frac{k_3}{\Delta t_1} - \frac{k_5}{\Delta t_3} + \frac{k_5 k_1}{\Delta t_3 \Delta t_1})u^{(0)} + (\frac{k_3}{\Delta t_1} - \frac{k_4}{\Delta t_2} - \frac{k_5 k_1}{\Delta t_3 \Delta t_1} + \frac{k_5 k_2}{\Delta t_3 \Delta t_2})u^{(1)} && (66)\\ &\quad + (\frac{k_4}{\Delta t_2} - \frac{k_2 k_5}{\Delta t_2 \Delta t_3})u^2 + \frac{k_5}{\Delta t_3}u^3 && (67) \end{aligned}$$

In the last step we used equation (58) to eliminate $u^{(2)}$. Again, we want positive coefficients for $u^{(0)}$, $u^{(1)}$, u^2, and u^3. This leads to the following inequalities,

$$\begin{aligned} \Delta t_3(\Delta t_1 - k_3) &\geq k_5(\Delta t_1 - k_1) && (68)\\ \Delta t_3(k_3\Delta t_2 - k_4\Delta t_1) &\geq k_5(k_1\Delta t_2 - k_2\Delta t_1) && (69)\\ \Delta t_3 k_4 &\geq k_2 k_5 && (70)\\ k_5 &\geq 0 && (71) \end{aligned}$$

Again, consider equation 2.17 of [37]. The second equation of equation 2.17 can be used to find $\beta_{32} = 1/(6\beta_{10}\beta_{21})$, and hence k_5. The fourth equation of equation 2.17 gives $k_3 + k_4 + k_5 = \Delta t$, while the third equation of equation 2.17 can be simplified to $(1/2 - P\beta_{32})/\beta_{10} = \beta_{31} + \alpha_{32}\beta_{21}$ which equals $k_4/\Delta t$. At this point one can write k_1, k_2, k_3, k_4, and k_5 in terms of Δt_1, Δt_2, m and n which then allows us to express inequalities (68) to (71) in terms of Δt_1, Δt_2, Δt_3, m and n. Then, inequalities (68) and (69) can be combined into the form

$$f(k, n) \leq m \leq 2 \tag{72}$$

where we plot f as a function of k for various n in Figure 27. As can be seen in the figure, the left hand side of inequality (72) is satisfied for all k and n when $m \geq 1.5$ but may be relaxed to smaller vales of m as k becomes larger and $n \to 1$. While n is one of our parameters, the value of $k = \Delta t_2/\Delta t_1$ can only be decreased since Δt_2 was already chosen to be an upper bound for positivity preservation. For $m \leq 2$, inequality (70) can be simplified to

$$\Delta t_3 \geq \frac{\Delta t_1}{2 - m} \tag{73}$$

which is a problematic condition that restricts the size of the third time step with a lower bound. Moreover, this condition suggests that m can be made as small as possible which contradicts with the left hand side of inequality (72). Note that one can always satisfy inequality (73) by clamping fluxes as discussed in Section 5, but this results in a loss of accuracy ameliorating some of the benefits of RK-3 over RK-2 to begin with. Even so, this clamped version of RK-3 may still have a better stability region.

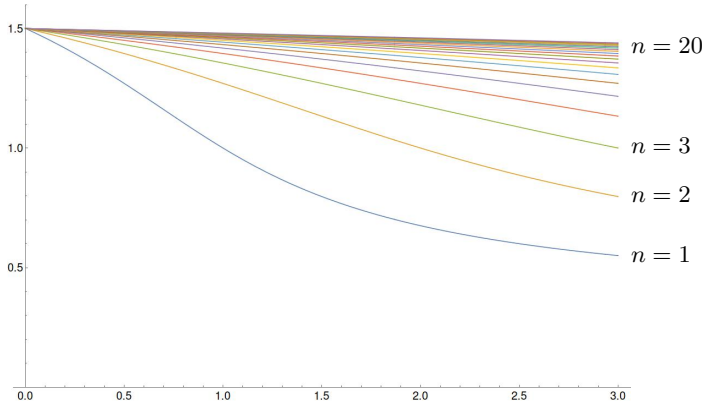


Figure 27: Plot of $f(k)$ for various values of n

In summary, Δt_1 is chosen based on the CFL condition and further clamped based on positivity. Δt_2 is chosen similarly determining the value of k. k can be made smaller if desired by reducing the value of Δt_2. Then $n = 1$ gives the best lower bound for m via inequality (72) and Figure 27. We choose $m = 1.5$ so the inequality (72) does not further restrict k. Finally we choose $\Delta t_3 = \Delta t_1/(2 - m) = 2\Delta t_1$ as a minimum allowable value according to inequality (73). Once all the parameters are chosen, equations (58) and (66) determine $u^{(3)}$. Table 4 shows the error and order of convergence of this scheme for $y' = -y$ where k is chosen randomly in the range $[0, 1]$ for each time step. Figure 28 shows the plot of the coefficients of $u^{(0)}$, $u^{(1)}$, u^2, and u^3 from equation (66) as a function of time for the case $\Delta t_1 = 0.01$. Note that the coefficients are always positive. Not exhaustively, but we have plotted coefficients for other choices of m and Δt_3 and seen that the coefficients become negative when either of the conditions in inequalities (72) or (73) is violated.

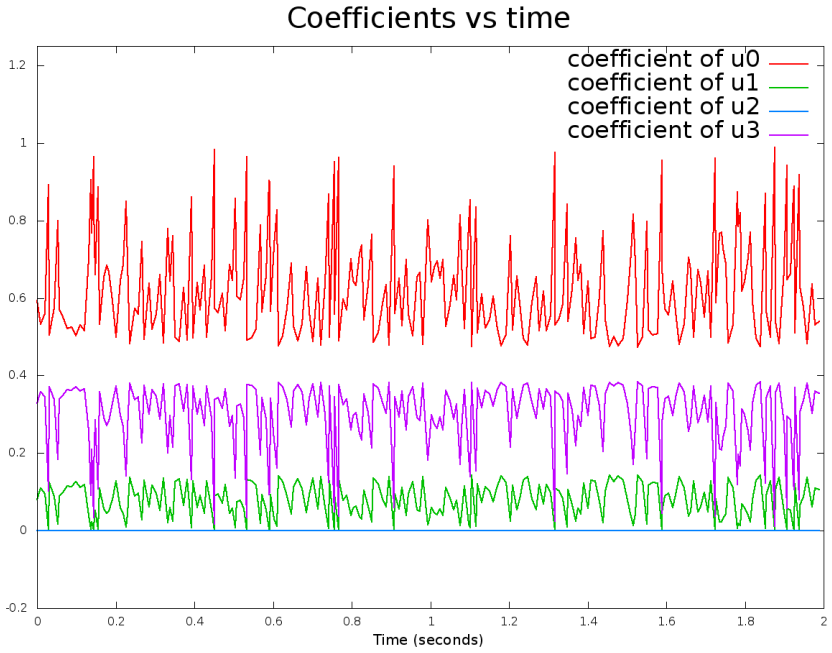


Figure 28: Plot of coefficients of $u^{(0)}$, $u^{(1)}$, u^2, and u^3 in equation (66) for $m = 1.5$ and $\Delta t_3 = \frac{\Delta t_1}{2-m}$ as a function of time when k is chosen randomly for each time step.

Δt_1 $\times 10^{-2}$	error	convergence order
1	1.01679e-08	-
.5	1.28509e-09	2.98408
.25	1.60989e-10	2.99684
.125	1.99881e-11	3.00975
.0625	2.52537e-12	2.98457
.03725	3.12889e-13	3.01277
.018625	3.94407e-14	2.98789

Table 4: Errors and order of accuracy for $y' = -y$ with the proposed TVD RK-3 scheme.