Journal of Computational Physics 228 (2009) 4146–4161

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A method for avoiding the acoustic time step restriction in compressible flow

Nipun Kwatra *, Jonathan Su, Jón T. Grétarsson, Ronald Fedkiw

Stanford University, 353 Serra Mall Room 207, Stanford, CA 94305, United States

ARTICLE INFO

Article history:
Received 16 July 2008
Received in revised form 12 February 2009
Accepted 19 February 2009
Available online 5 March 2009

Keywords:
Compressible flow
Incompressible flow
Conservative scheme
Pressure projection
Shock capturing

ABSTRACT

We propose a novel method for alleviating the stringent CFL condition imposed by the sound speed in simulating inviscid compressible flow with shocks, contacts and rarefactions. Our method is based on the pressure evolution equation, so it works for arbitrary equations of state, chemical species etc. and is derived in a straight-forward manner. Similar methods have been proposed in the literature, but the equations they are based on and the details of the methods differ significantly. Notably our method leads to a standard Poisson equation similar to what one would solve for incompressible flow, but has an identity term more similar to a diffusion equation. In the limit as the sound speed goes to infinity, one obtains the Poisson equation for incompressible flow. This makes the method suitable for two-way coupling between compressible and incompressible flows and fully implicit solid–fluid coupling, although both of these applications are left to future work. We present a number of examples to illustrate the quality and behavior of the method in both one and two spatial dimensions, and show that for a low Mach number test case we can use a CFL number of 300 (whereas previous work was only able to use a CFL number of 3 on the same example).

1. Introduction

In this paper, we focus on highly non-linear compressible flows with shocks, contacts and rarefactions, for example the Sod shock tube. Traditionally these types of problems are solved with explicit time integration (Runge–Kutta methods, ENO, WENO etc. see e.g. [10,11,5]). Although these methods produce high quality results, small time steps are required in order to enforce the CFL condition of information moving only one grid cell per time step. While this is understandable for very high Mach number flow where $|u|, |u-c|$ and $|u+c|$ are all of similar magnitude, it is too restrictive for flows where the sound speed, c, may be much larger than $|u|$. Moreover some flow fields might have both high Mach number regions where shock waves are of interest as well as low Mach number regions where the material velocities are important. In this case, a large number of time steps are required if one is interested in the motion of the fluid particles over an appreciable distance in the low Mach number regions. Thus, it can be quite useful to have methods that avoid the stringent CFL time step restriction imposed by the acoustic waves and instead use only the material velocity CFL restriction (albeit one would expect some loss of quality because of the implicit treatment of the acoustic waves).

To alleviate the stringent CFL restriction, [6] proposed both a non-conservative and a conservative scheme. Their non-conservative scheme builds on the predictor–corrector type scheme of [16] to derive an elliptic pressure equation quite similar to ours, but for an adiabatic fluid. Our method is similar in spirit to [6,13–15] where the calculation is divided into

* Corresponding author. Tel.: +1 6506195898.
E-mail addresses: kwatra@stanford.edu (N. Kwatra), jonsu@stanford.edu (J. Su), jontg@stanford.edu (J.T. Grétarsson), fedkiw@cs.stanford.edu (R. Fedkiw).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.02.027

two parts: advection and non-advection. The advection terms are treated with explicit time integration, and thus the CFL restriction on the material velocity remains. Whereas one can use a standard method such as ENO in solving the advection terms, we found that when coupled to an implicit solution of the pressure equations (that is inherently central-differenced) the standard ENO method sometimes leads to spurious oscillatory behavior. Thus we designed a new ENO method geared towards a MAC grid discretization of the data, making it more similar to incompressible flow. We call this MAC-ENO or MENO. The remaining non-advection terms are solved using an implicit equation for the pressure using a standard MAC grid type formulation. Since the MAC grid is dual in both velocity and pressure (noting that the MAC grid pressure needs to live at cell faces for flux based methods), one needs to interpolate data back and forth.

We base the derivation of our method on the pressure evolution equation as discussed in [2], thus making it valid for general equations of state, arbitrary chemical species etc. Thus, our derivation has fewer assumptions and is more straight-forward than previous work, especially those based on preconditioners. For example, [13] makes two critical assumptions in their derivation of the implicit equation for pressure. In approximating the derivative of momentum they discard a $\Delta t \frac{\nabla p}{\rho}$ term, and their pressure evolution equation is missing the advection term. Also, our method is fully conservative and thus shocks are tracked at the right speed. We present a number of traditional examples for highly non-linear compressible flows including the Sod shock tube, interacting blast waves, and in two dimensions we show Flow Past a Step, Double Mach Reflection of a Strong Shock and a Circular Shock. We also demonstrate that the method works well for low Mach number flow, taking an example from [7] where the authors obtain reasonable results with a CFL number of 3. Notably, our method allows a CFL number of 300 (two orders of magnitude more).

2. Numerical method

Let us consider the one dimensional Euler equations,

$$\begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ Eu + pu \end{pmatrix}_x = 0,$$

with ρ being the density, u the velocity, E the total energy per unit volume and p the pressure. The flux term can be separated into an advection part and a non-advection part,

$$\mathbf{F}_1(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 \\ Eu \end{pmatrix}, \quad \mathbf{F}_2(\mathbf{U}) = \begin{pmatrix} 0 \\ p \\ pu \end{pmatrix}. \tag{1}$$

We first compute the Jacobian of the advection part

$$\mathbf{J} = \begin{pmatrix} 0 & 1 & 0 \\ -u^2 & 2u & 0 \\ -\frac{Eu}{\rho} & \frac{E}{\rho} & u \end{pmatrix}.$$

All the Jacobian's eigenvalues are equal to u, and it is rank deficient with left eigenvectors of $(u, -1, 0)$ and $(E/\rho, 0, -1)$ and right eigenvectors of $(1, u, 0)^T$ and $(0, 0, 1)^T$. Since all the characteristic velocities are identical, we can apply component wise upwinding to $\mathbf{F}_1(\mathbf{U})$ without having to transform into the characteristic variables first (as in [4]). Moreover, this advection part only requires a time step restriction based on u.

2.1. Implicit pressure update

The multi-dimensional Euler equations are

$$\begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 \\ \rho uv \\ \rho uw \\ Eu \end{pmatrix}_x + \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 \\ \rho vw \\ Ev \end{pmatrix}_y + \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 \\ Ew \end{pmatrix}_z + \begin{pmatrix} 0 \\ \nabla p \\ \nabla \cdot (p\vec{u}) \end{pmatrix} = 0,$$

where $\vec{u} = (u, v, w)$ are the velocities. Here we have advection components in each of the three spatial dimensions, and they can be handled as outlined previously in a dimension by dimension fashion (as in [11]).

We apply a time splitting as is typical for incompressible flow formulations, first updating the advection terms to obtain an intermediate value of the conserved variables $(\rho)^*, (\rho u)^*$ and E^*, and afterward correct these to time t^{n+1} using an implicit pressure. Since the pressure does not affect the continuity equation, $\rho^{n+1} = \rho^*$. The non-advection momentum and energy updates are

$$\frac{(\rho\vec{u})^{n+1} - (\rho\vec{u})^*}{\Delta t} = -\nabla p \tag{2}$$

and

$$\frac{E^{n+1} - E^*}{\Delta t} = -\nabla \cdot (pu). \tag{3}$$

Taking motivation from the standard incompressible flow formulation (which uses the momentum equation to derive an implicit equation for pressure), we divide Eq. (2) by ρ^{n+1},

$$\vec{u}^{n+1} = \vec{u}^\star - \Delta t \frac{\nabla p}{\rho^{n+1}} \tag{4}$$

and take its divergence to obtain

$$\nabla \cdot \vec{u}^{n+1} = \nabla \cdot \vec{u}^\star - \Delta t \nabla \cdot \left(\frac{\nabla p}{\rho^{n+1}}\right). \tag{5}$$

In the case of incompressible flow, we would set $\nabla \cdot \vec{u}^{n+1} = 0$, but for compressible flow we instead use the pressure evolution equation derived in [2],

$$p_t + \vec{u} \cdot \nabla p = -\rho c^2 \nabla \cdot \vec{u}. \tag{6}$$

If we fix $\nabla \cdot \vec{u}$ to be at time $n+1$ through the time step (making an $\mathcal{O}(\Delta t)$ error), we can substitute in Eq. (5) to get

$$p_t + \vec{u} \cdot \nabla p = -\rho c^2 \nabla \cdot \vec{u}^\star + \rho c^2 \Delta t \nabla \cdot \left(\frac{\nabla p}{\rho^{n+1}}\right), \tag{7}$$

which is an advection–diffusion equation with a source term. Discretizing the $\vec{u} \cdot \nabla p$ advection term explicitly, using a forward Euler time step, and defining the diffusive pressure at time t^{n+1} as is typical for backward Euler discretization, gives after rearrangement

$$p^{n+1} - \rho^n (c^2)^n \Delta t^2 \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho^{n+1}}\right) = (p^n - (\vec{u}^n \cdot \nabla p^n)\Delta t) - \rho^n (c^2)^n \Delta t \nabla \cdot \vec{u}^\star. \tag{8}$$

Note we have discretized ρc^2 at time t^n. This equation can be further simplified by using the advection equation for pressure,

$$\frac{p^a - p^n}{\Delta t} + \vec{u}^n \cdot \nabla p^n = 0$$

to obtain

$$p^a = p^n - (\vec{u}^n \cdot \nabla p^n)\Delta t, \tag{9}$$

where p^a is an advected pressure which can be computed using HJ ENO [9] or semi-Lagrangian advection [1]. Substituting in Eq. (8) we obtain

$$p^{n+1} - \rho^n (c^2)^n \Delta t^2 \nabla \cdot \left(\frac{\nabla p^{n+1}}{\rho^{n+1}}\right) = p^a - \rho^n (c^2)^n \Delta t \nabla \cdot \vec{u}^\star. \tag{10}$$

We discretize this equation at cell centers (which is typical for advection–diffusion equations) and thus need to define velocities at cell faces for $\nabla \cdot \vec{u}^\star$. Consider two adjacent grid cells, one centered at X_i and one centered at X_{i+1}. We divide these into four regions $C_{i,L}, C_{i,R}, C_{i+1,L}, C_{i+1,R}$, where $(C_{i,R} \cup C_{i+1,L})$ represents a dual cell (see Fig. 1). Then Eq. (2) for $C_{i,R}$ is

$$\frac{(\rho u)^{n+1}_{i,R} - (\rho u)^*_{i,R}}{\Delta t} = -\frac{p^{n+1}_{i+1/2} - p^{n+1}_i}{\Delta x/2}. \tag{11}$$

Similarly for $C_{i+1,L}$ we have

$$\frac{(\rho u)^{n+1}_{i+1,L} - (\rho u)^*_{i+1,L}}{\Delta t} = -\frac{p^{n+1}_{i+1} - p^{n+1}_{i+1/2}}{\Delta x/2}. \tag{12}$$

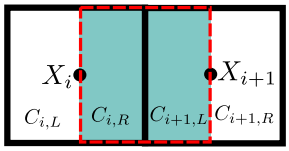


Fig. 1. Two neighboring cells with the dual cell shaded.

Adding these equations together and dividing by $(\rho_i + \rho_{i+1})$ yields

$$\frac{\hat{u}^{n+1}_{i+1/2} - \hat{u}^{*}_{i+1/2}}{\Delta t} = -\frac{p^{n+1}_{i+1} - p^{n+1}_{i}}{\Delta x \hat{\rho}^{n+1}}, \tag{13}$$

where $\hat{u}_{i+1/2} = \frac{(\rho u)_{i,R} + (\rho u)_{i+1,L}}{\rho_i + \rho_{i+1}} = \frac{(\rho u)_i + (\rho u)_{i+1}}{\rho_i + \rho_{i+1}}$ can be thought of as a density-weighted face velocity, and $\hat{\rho}_{i+1/2} = \frac{\rho_i + \rho_{i+1}}{2}$ is the cell face density. Note that we currently use $(\rho u)_{i,R} = (\rho u)_i$ and $(\rho u)_{i+1,L} = (\rho u)_{i+1}$, although higher order approximations could be used. Using this discretization on Eq. (10) yields

$$\left[I + \rho^n (c^2)^n \Delta t^2 G^T \left(\frac{1}{\tilde{\rho}^{n+1}} G\right)\right] p^{n+1} = p^a + \rho^n (c^2)^n \Delta t G^T \vec{u}^{\star}, \tag{14}$$

where G is our discretized gradient operator and $-G^T$ is our discretized divergence operator. This is solved to obtain p^{n+1} at cell centers.

In summary, instead of using an equation of state (EOS) to find the pressure for use as a flux in both conservation of momentum and energy, we use Eq. (14). The EOS still plays a role because it is used to determine the time t^n pressures which factor into p^a and is also used to determine $(c^2)^n$. In Fig. 2 we show an example calculation of the pressure for our Sod shock tube example. In the picture we plot the pressure using the equation of state at time t^n, i.e. p^n, the pressure calculated using Eq. (14), i.e. our p^{n+1}, and also the pressure calculated using the EOS applied to the conservative variables at time t^{n+1}, i.e. p^{n+1}_{EOS}. Notice in the figure that the pressure calculated from Eq. (14) is a good approximation to what the pressure will be at the next time step (i.e. p^{n+1}_{EOS}) emphasizing the implicit nature of our scheme. p^n is the pressure used in a typical explicit scheme.

It is interesting to note that this derivation does not require an ideal gas assumption, and hence should be general enough to work with any EOS (even multi-species flow [2]).

2.2. Updating momentum and energy

To obtain the correct shock speeds we use a flux based method and thus need the pressure at cell faces for Eqs. (2) and (3), and the velocity at cell faces for Eq. (3). Applying conservation of momentum to the control volumes $C_{i,R}$ and $C_{i+1,L}$ (see Fig. 1) gives

$$Du_{i,R}/Dt = (p_i - p_{i+1/2})/(\Delta x \rho_{i,R}/2)$$

and

$$Du_{i+1,L}/Dt = (p_{i+1/2} - p_{i+1})/(\Delta x \rho_{i+1,L}/2).$$

The constraint that the interface remain in contact implies that $Du_{i,R}/Dt = Du_{i+1,L}/Dt$, which can be used with the aforementioned equations to solve for the pressure at the flux location $X_{i+1/2}$ as

$$p_{i+1/2} = \frac{p_{i+1}\rho_i + p_i \rho_{i+1}}{\rho_{i+1} + \rho_i}. \tag{15}$$

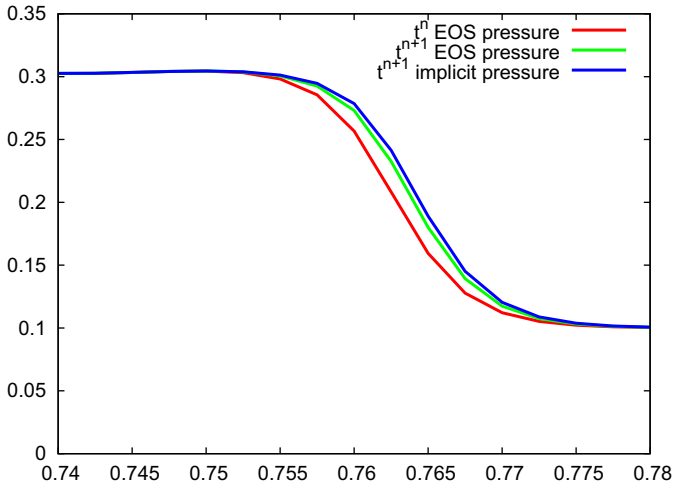


Fig. 2. A blow-up of the pressure plot for example 6.1.1 at time $t(n) = .149$ s and $t(n+1) = .15$ s, showing that the implicit pressure calculated in Eq. (14) is a good approximation to what the pressure will be at time t^{n+1} emphasizing the implicit nature of our scheme. p^n is also plotted to emphasize the difference between using an implicit and explicit pressure.

For solid wall boundaries, we reflect the pressure and density values as usual, and then use Eq. (15). The cell face velocity is computed via Eq. (13), and $p_{i+1/2}\hat{u}_{i+1/2}$ is used in Eq. (3).

3. Time step restriction

The eigenvalues of the Jacobian of the advection part of the flux are all u. Since we solve the acoustic component implicitly, we no longer have a severe time step restriction determined by the speed of sound c, and all that remains is to find an estimate for the maximum value of $|u|$ throughout the time step. Simply using u^n is not enough, since e.g. Sod shock tube starts out with an initial velocity identically zero and thus u^n would imply an infinite Δt. To alleviate this, we add a term that estimates the change in velocity over a time step similar to what was done in [8]. Assuming the flow is smooth, we combine conservation of mass and momentum to give an equation for the velocity, $u_t + u \cdot \nabla u + \frac{\nabla p}{\rho} = 0$. The temporal update of this equation would advect velocity based on the $u \cdot \nabla u$ term, but also increase the velocity by an amount equal to $\frac{\nabla p}{\rho}$. In one spatial dimension, we use this to estimate the velocity at the end of the time step as $\left(\frac{|u^n|_{max} + \frac{|p_x|}{\rho}\Delta t}{\Delta x}\right)$ and the CFL condition becomes

$$\Delta t\left(\frac{|u^n|_{max} + \frac{|p_x|}{\rho}\Delta t}{\Delta x}\right) \leqslant 1. \tag{16}$$

This is quadratic in Δt with solutions

$$\frac{-|u^n|_{max} - \sqrt{|u^n|^2_{max} + 4\frac{|p_x|}{\rho}\Delta x}}{2|p_x|/\rho} \leqslant \Delta t \leqslant \frac{-|u^n|_{max} + \sqrt{|u^n|^2_{max} + 4\frac{|p_x|}{\rho}\Delta x}}{2|p_x|/\rho}.$$

As the lower limit is always non positive and $\Delta t \geqslant 0$, we only need to enforce the upper bound. As $p_x \to 0$, both the numerator and denominator vanish and thus we obtain a more convenient time step restriction by replacing the $2nd$ Δt in Eq. (16) with this upper bound to obtain

$$\frac{\Delta t}{2}\left(\frac{|u^n|_{max}}{\Delta x} + \sqrt{\left(\frac{|u^n|_{max}}{\Delta x}\right)^2 + 4\frac{|p_x|}{\rho\Delta x}}\right) \leqslant 1. \tag{17}$$

Note that this is not linear in Δx, but as $\Delta x \to 0$ we obtain a more typical CFL condition $\Delta t < \frac{\Delta x}{|u^n|_{max}}$. In two spatial dimensions our CFL follows along the lines of [8]'s equation 95 and is given by:

$$\frac{\Delta t}{2}\left(\frac{|u|_{max}}{\Delta x} + \frac{|v|_{max}}{\Delta y} + \sqrt{\left(\frac{|u|_{max}}{\Delta x} + \frac{|v|_{max}}{\Delta y}\right)^2 + 4\frac{|p_x|}{\rho\Delta x} + 4\frac{|p_y|}{\rho\Delta y}}\right) \leqslant 1.$$

All of our examples are stable for CFL number $\alpha = .9$, and *all* of our examples were unstable for $\alpha = 1.3$. Some examples (e.g. example 6.1.8) blow up for $\alpha = 1$.

4. Modified ENO scheme

When using traditional ENO methods for the advection part of our equations (as in [11]), we obtained excessive spurious oscillations. This seems to be related to our dual cell center and MAC grid formulation, thus we device a new ENO scheme which better utilizes that dual formulation. We call this Mach-ENO or MENO. The main idea is to replace the advection velocity with the MAC grid value defined at the flux in question, i.e. $\hat{u}$. The lowest level of the divided difference table is typically constructed with the physical fluxes, i.e. $\rho u, \rho u^2$ and Eu for $\mathbf{F}_1(\mathbf{U})$ in Eq. (1). A dissipation term is added for the local and global Lax-Friedrichs versions. Consider constructing an ENO approximation for the flux at $X_{i+1/2}$. Locally, we would use a divided difference table with base values corresponding to the physical fluxes plus or minus the appropriate dissipation. Our modification is to replace $\rho_j u_j, \rho_j u_j^2$ and $E_j u_j$ with $\rho_j \hat{u}_{i+1/2}, \rho_j u_j \hat{u}_{i+1/2}$ and $E_j \hat{u}_{i+1/2}$ leaving the dissipation terms unaltered. Note that $\hat{u}_{i+1/2}$ is fixed throughout the divided difference table similar to the way one fixes the dissipation coefficient.

In order to validate our new MENO scheme, we compared it to the standard scheme from [11] for the standard Sod shock tube in Fig. 3. For this problem and other fully explicit simulations the results were fairly similar, but when we ran the simulations with our semi-implicit formulation the MENO scheme performed much better, and in fact the standard ENO scheme was not successful in producing any solution whatsoever for Fig. 11 in our examples section.

5. Time integration

While the explicit component of our update is an upwind scheme, the implicit component is centrally-differenced. This tends to introduce more dispersive rather than dissipative errors to the solution (i.e. there is more of an imaginary component to the eigenvalues), which suggests the use of Runge–Kutta over forward Euler.

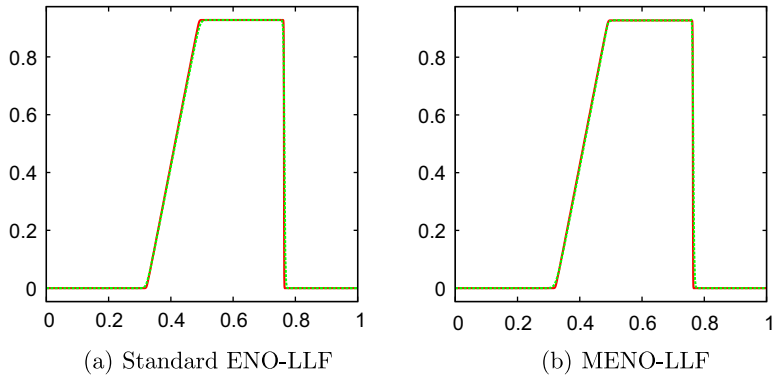


Fig. 3. Sod shock tube problem at $t = .15$ s. Left: Standard ENO-LLF (Local Lax-Friedrichs) using 401 grid points (green) and 1601 grid points (red). Right: The base 1601 grid points solution is the same as in the left figure, but the coarse grid calculation (with 401 grid points) is done with the new MENO scheme. Velocity is shown in both figures. Both simulations were done with explicit time integration and a full characteristic decomposition in order to demonstrate that the new ENO schemes performs similar to the old one when one is not using our new implicit discretization of the pressure. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

We use two variations of the third order TVD Runge–Kutta scheme [10] in all of our examples. The first is to perform Runge–Kutta on just the advection part, $\mathbf{F}_1(\mathbf{U})$, with only one final implicit solve for $\mathbf{F}_2(\mathbf{U})$. The second variation is to carry out both $\mathbf{F}_1(\mathbf{U})$ and $\mathbf{F}_2(\mathbf{U})$ for each Runge–Kutta stage, noting that this has three times the computational cost as far as the implicit solution of $\mathbf{F}_2(\mathbf{U})$ is concerned. In general we observed better performance, especially in controlling overshoots, when using the second variation (see Fig. 4). However, some examples (in particular the high Mach number ones) do tend to show more oscillations (see Fig. 4, bottom). These oscillations are less predominant when combined with MENO, so we show all of our examples with the second variation.

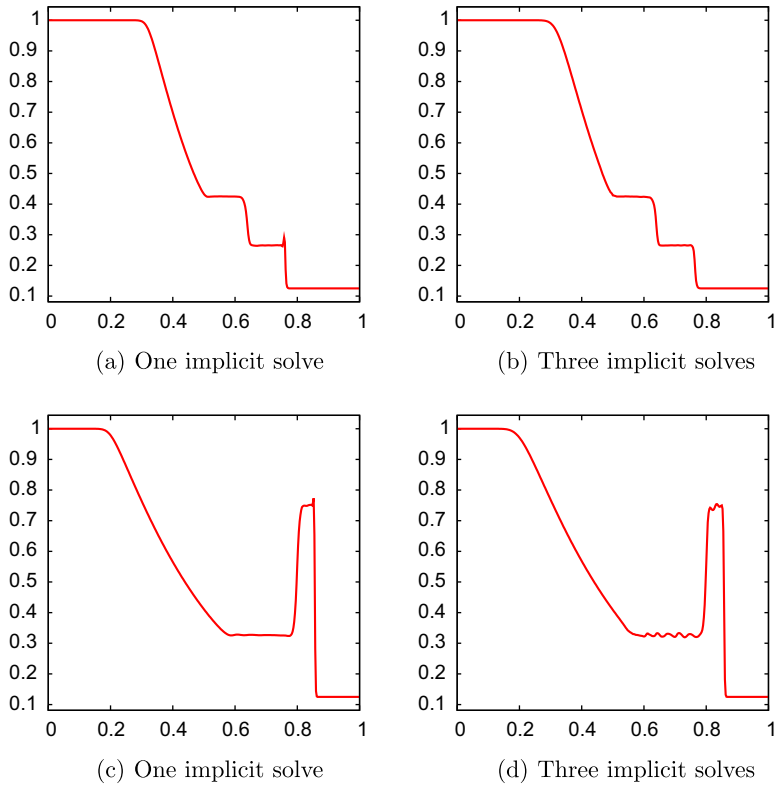


Fig. 4. Numerical results comparing placing the implicit solve either inside each Runge–Kutta stage (b and d) or once after a full three stage Runge–Kutta cycle (a and c). The top two figures show the results for a Sod shock tube problem at $t = .15$ s, the bottom two figures show the results for a strong shock tube problem at $t = 2.5 \times 10^{-6}$ s. Density is shown in all figures. Note the spurious overshoots when the implicit solve is not included in the Runge–Kutta cycle (left two figures). Note that we use the standard ENO scheme from [11] (not MENO) for these four examples.

6. Numerical results

6.1. One dimensional validation

For the one dimensional tests, we use a computational domain of [0,1], 401 grid points, and also plot a baseline solution using 1601 grid points in the standard fully explicit ENO method as in [11]. A second order ENO was used along with the CFL number of .5. Unless otherwise noted the maximum Mach number in each example lies within the range (.9,2.5). All units are in S.I. Generally speaking our method is a perturbation of those proposed by [13,14] and thus demonstrates similar qualitative behavior. Timings are shown in Table 1. In particular note that the implicit scheme is generally more efficient than the explicit scheme predominantly because we avoid the characteristic decomposition and can advect all three independent variables simultaneously because they all have the same eigenvalue u. At first glance one might assume that the necessity of a pressure Poisson equation would cancel out these efficiency gains, but practical experience shows only five or six iterations of conjugate gradients is required to reach a reasonable tolerance. It is unclear whether our newly proposed semi-implicit method would have these slight efficiency gains across a wider number of examples and in multiple spatial dimensions, however for the low Mach number flow problems for which it was designed (such as example 6.1.8) it is significantly more efficient than the explicit method.

Table 1
Wall clock times comparing the semi-implicit method with the fully explicit method, for 1-D examples. Simulations were run to the target times of each example as mentioned in their respective figures.

Test name	Semi-implicit (s)	Explicit (s)
Sod shock tube	2.95	3.69
Lax shock tube	2.71	4.53
Strong shock tube	2.43	3.43
Mach 3 shock test	2.90	3.59
High Mach flow test	3.75	3.29
Interaction of blast waves (Bang Bang)	5.28	9.86
Two symmetric rarefaction waves	3.52	4.15

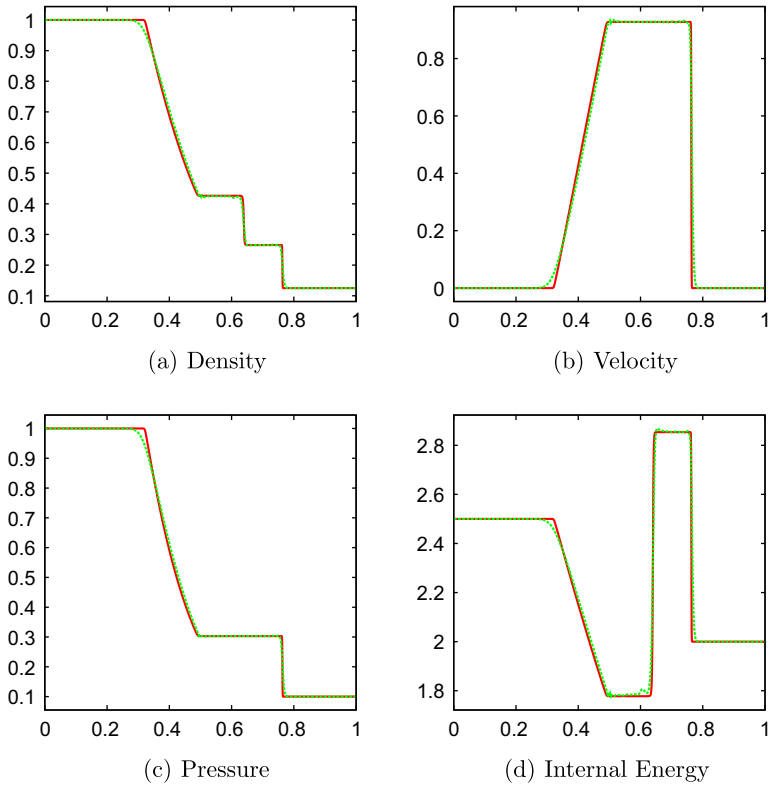


Fig. 5. Numerical results of the Sod shock tube problem at $t = .15$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

6.1.1. Sod shock tube

Our first test case is a standard Sod shock tube with initial conditions of

$$(\rho(x,0),u(x,0),p(x,0)) = \begin{cases} (1,0,1), & \text{if } x \leqslant .5, \\ (.125,0,.1), & \text{if } x > .5. \end{cases}$$

Our results are shown in Fig. 5, which indicate well resolved shock, rarefaction and contact solutions. Since our method is conservative, we get the correct shock speeds. The results are comparable to that of [7,13].

6.1.2. Lax's shock tube

Lax's shock tube is similar in nature to Sod shock tube, except that the initial condition has a discontinuity in the velocity:

$$(\rho(x,0),u(x,0),p(x,0)) = \begin{cases} (.445,.698,3.528), & \text{if } x \leqslant .5, \\ (.5,0,.571), & \text{if } x > .5. \end{cases}$$

Our results are shown in Fig. 6. Again, the results are comparable to the previous work.

6.1.3. Strong shock tube

The Strong shock tube problem poses initial conditions that generates a supersonic shock:

$$(\rho(x,0),u(x,0),p(x,0)) = \begin{cases} (1,0,10^{10}), & \text{if } x \leqslant .5, \\ (.125,0,.1), & \text{if } x > .5. \end{cases}$$

Our results are shown in Fig. 7. The scheme admits some oscillations near the rarefaction wave, and we see no notable difference in simulation time when compared to the explicit simulation. With that in mind, we note that the main advantage of the proposed method is to take time steps irrespective of the sound speed values; in cases of high Mach number flows (or high Mach number regions of the flow – if asynchronous time integration is used), one could use a typical ENO scheme.

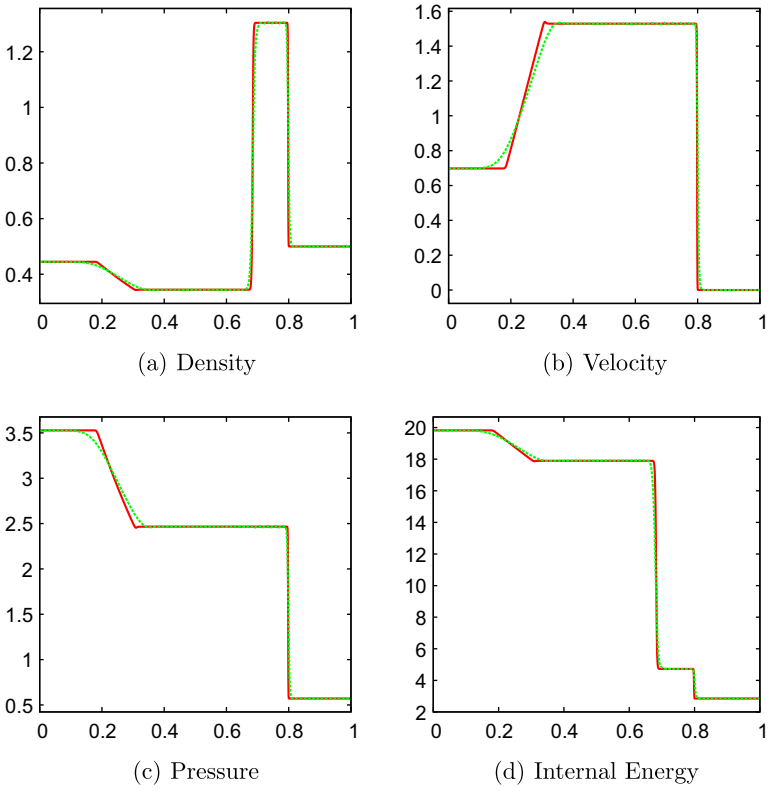


Fig. 6. Numerical results of the Lax's shock tube problem at $t = .12$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

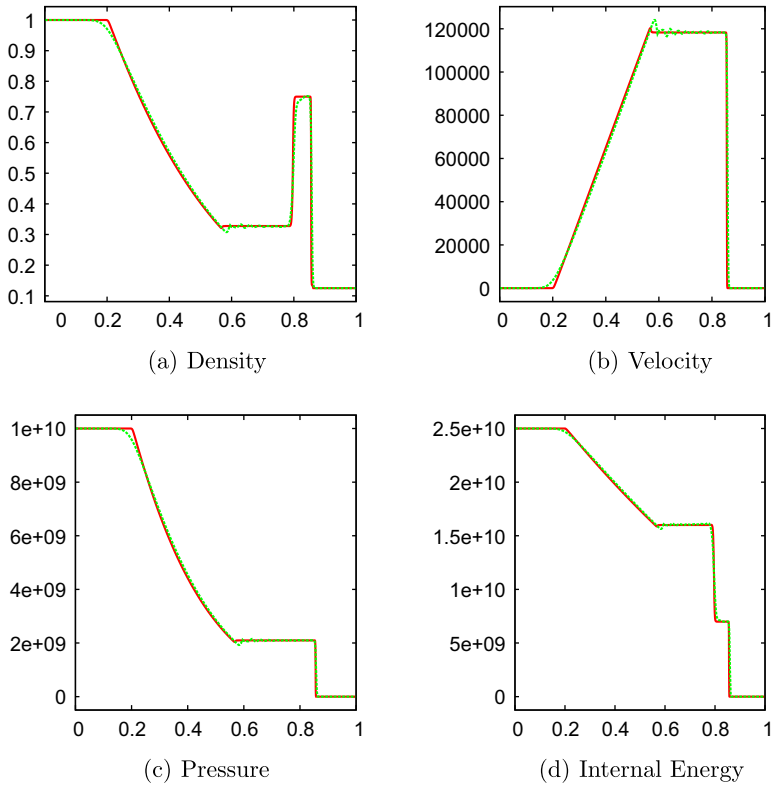


Fig. 7. Numerical results of the strong shock tube problem at $t = 2.5 \times 10^{-6}$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

6.1.4. Mach 3 shock test

The initial conditions for the Mach 3 shock test are:

$$(\rho(x,0), u(x,0), p(x,0)) = \begin{cases} (3.857, .92, 10.333), & \text{if } x \leqslant .5, \\ (1, 3.55, 1), & \text{if } x > .5. \end{cases}$$

Our results are shown in Fig. 8. As above we do note some oscillations near the rarefaction wave.

6.1.5. High mach flow test

The initial conditions for the High mach flow test are:

$$(\rho(x,0), u(x,0), p(x,0)) = \begin{cases} (10, 2000, 500), & \text{if } x \leqslant .5, \\ (20, 0, 500), & \text{if } x > .5. \end{cases}$$

As noted in [7] the Mach number in this test can reach as high as 240. Our results are shown in Fig. 9.

6.1.6. Interaction of blast waves

Here we present a test of two interacting blast waves. This problem was introduced by [12] and involves multiple strong shock waves. The initial conditions for the test are:

$$(\rho(x,0), u(x,0), p(x,0)) = \begin{cases} (1, 0, 10^3), & \text{if } 0 \leqslant x < .1, \\ (1, 0, 10^{-2}), & \text{if } .1 \leqslant x < .9, \\ (1, 0, 10^2), & \text{if } .9 \leqslant x \leqslant 1. \end{cases}$$

We also have solid wall boundary conditions at $x = 0$ and $x = 1$. Our results are shown in Fig. 10 which shows that we achieve very accurate results.

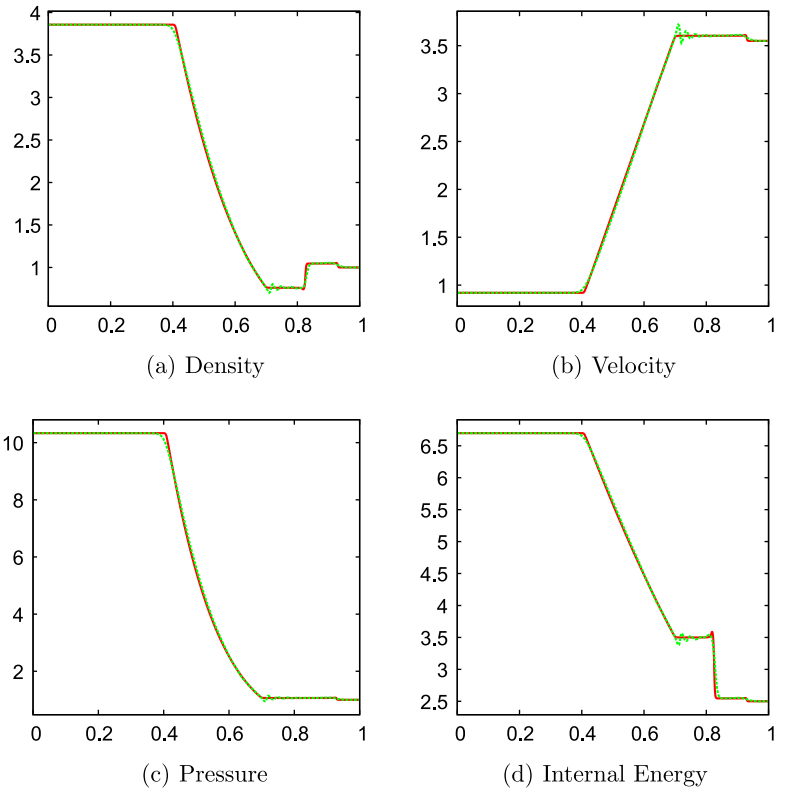


Fig. 8. Numerical results of the Mach 3 shock tube problem at $t = .09$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

6.1.7. Two symmetric rarefaction waves

In this test there are two rarefaction waves going in opposite directions from the center of the domain. This causes very low density regions near the center of the domain. The initial conditions for the test are:

$$(\rho(x,0), u(x,0), p(x,0)) = \begin{cases} (1,-2,.4), & \text{if } x \leqslant .5, \\ (1,2,.4), & \text{if } x > .5. \end{cases}$$

Our results are shown in Fig. 11. Our results are comparable to that of [7,13]. Note that there is an unphysical pulse in the internal energy field near the low pressure region, caused by overheating (see e.g. [3]).

6.1.8. Smooth flow test (mach zero limit)

The initial conditions for the zero mach limit test are given by:

$$\begin{aligned} u(x,0) &= 0, \\ p(x,0) &= p_0 + \epsilon p_1(x), \\ p_1(x) &= 60\cos(2\pi x) + 100\sin(4\pi x), \\ \rho(x,0) &= \left(\frac{p(x,0)}{p_0}\right)^{\frac{1}{\gamma}} \rho_0, \end{aligned}$$

where $\rho_0 = 1, p_0 = 10^9$ and $\epsilon = 10^3$. Since the flow is smooth and there are no shocks in this test, we have used a single implicit solve per time step. This test is dominated by acoustic waves (as observed in [7]). We can take time steps as large as is permitted by our CFL condition in Eq. (17). This permits time steps three orders of magnitude greater than those permitted by sound speed based CFL. However, as with all implicit schemes, taking too large a time step can lead to inaccurate results. Thus, in order to get sufficient accuracy, we clamp our time step to be a fixed multiple of the explicit time step (which is calculated using the sound speed based CFL). In Fig. 12 we use three times the explicit time step and show convergence via grid resolution.

In a second suit of tests we show that we can increase the grid resolution without the need to refine the time step. The timing results for this experiment are available in Table 2, where Δt remains fixed as the grid resolution goes up as high as

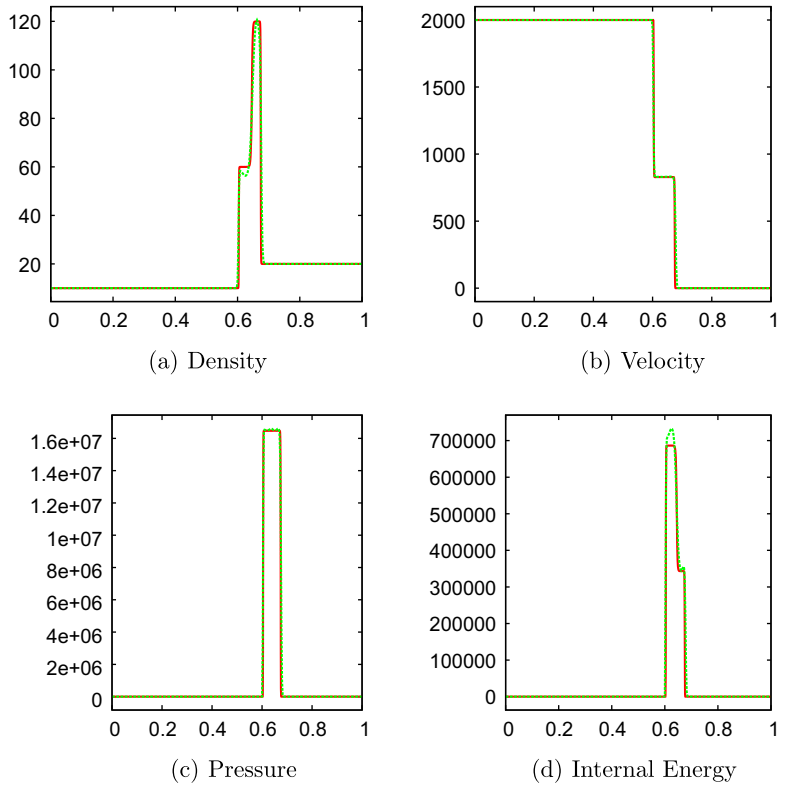


Fig. 9. Numerical results of the High Mach shock tube problem at $t = 1.75 \times 10^{-4}$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

320,000 grid cells. At that point the effective sound speed CFL is 300. Numerical results are plotted in Fig. 13 and Table 2 summarizes the results. In particular we note that the newly proposed implicit method permits a fixed time step all the way up to 320,000 grid points. This allows the wall clock simulation time to scale approximately linear to the size of the problem (since we solve the Poisson equation using conjugate gradients, which has superlinear complexity – however, note that one only needs the solver to converge in the sense of truncation error as opposed to round-off error). On the other hand, in explicit methods the simulation time grows quadratically, becoming impractical at 320,000 grid points. Note that since we are not refining the time step, we do not expect to see any further convergence in the solution.

6.2. Flow past a step test

Our first two-dimensional experiment is similar to the one described in [3]. We assume an ideal gas with $\gamma = 1.4$. The test domain is 3 units long and 1 unit wide, with a .2 unit high step which is located .6 units from the left hand side of the tunnel. The initial conditions are $\rho = 1.4, p = 1$ and $u = 3$ and $v = 0$ everywhere in the domain. We apply an inflow boundary condition on the left hand side of the domain, and an outflow boundary condition on the right hand side of the domain. A reflective solid wall boundary condition is applied for the top and bottom boundaries of the domain. We show numerical results at $t = 4$ s on a grid of resolution 120×40 in Fig. 14.

6.3. Double mach reflection of a strong shock

In a computational domain of $[0,4] \times [0,1]$, a planar Mach 10 shock hits a reflecting boundary that lies along the bottom wall of the domain along $x \in [\frac{1}{6}, 4]$. The plane of the shock begins at $(\frac{1}{6}, 0)$ and makes a 60° angle with the reflecting plane. The left and bottom (for $x \in (0, \frac{1}{6})$) boundary conditions are given by the postshock condition, the right boundary by a zero-gradient condition, and the top boundary is set to describe the exact motion of the Mach 10 shock. If we take $\vec{n}$ to be the unit vector that lies normal to the planar shock, then the initial values are given by:

$$(\rho(x,y,0), u(x,y,0), p(x,y,0)) = \begin{cases} (1.4, \vec{0}, 1), & \text{preshock}, \\ (8, 8.25\vec{n}, 116.5), & \text{postshock}. \end{cases}$$

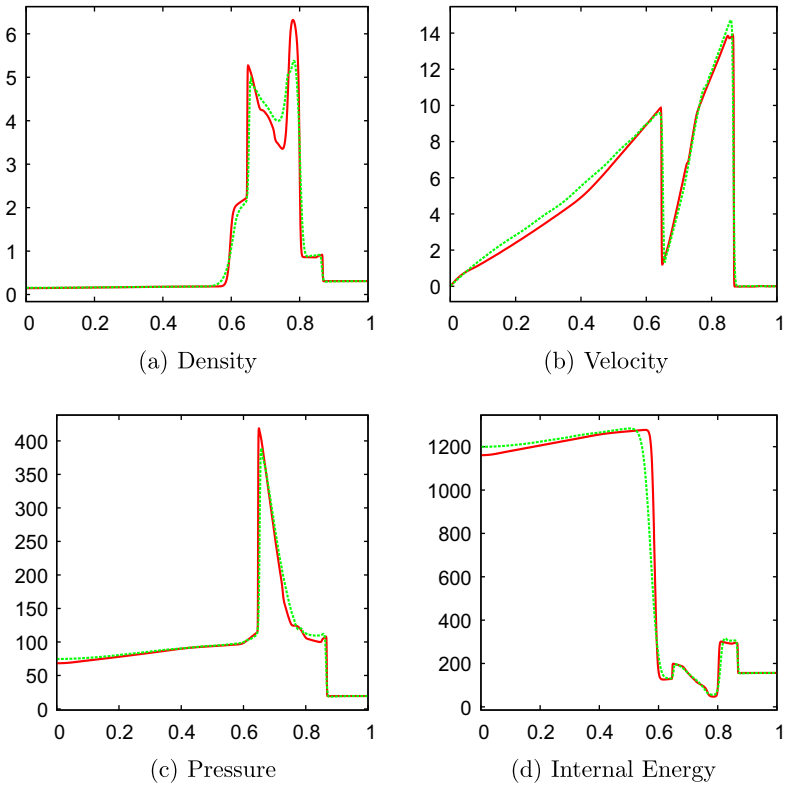


Fig. 10. Numerical results of the interacting blasts shock tube problem at $t = .038$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

Our method (see Fig. 15) compares well with those provided in [12], which provides a description of this example and presents numerical results comparing the performance of various methods in this problem. As is done in previous work we only show the domain of interest ($[0,3] \times [0,1]$).

6.4. Circular shock test

The circular shock test has an initial condition prescribed as

$$(\rho, u, v, p) = \begin{cases} (1,0,0,1), & \text{if } r \leqslant .4, \\ (.125,0,0,.1), & \text{if } r > .4, \end{cases}$$

where $r = \sqrt{x^2 + y^2}$. Numerical results are shown in Fig. 16. The same test was shown in [14]. Our results indicate well resolved shock and contact solutions along with correct speed shock calculations.

7. Conclusions and future work

We have presented a method for alleviating the stringent CFL condition imposed by the sound speed in highly non-linear compressible flow simulations. A fractional step procedure combined with the pressure evolution equation is used. The method works for arbitrary equations of state, and in the limit as the sound speed goes to infinity it yields the Poisson equation for incompressible flow. We also presented a Mach-ENO or MENO scheme which better utilizes a dual cell center and MAC grid formulation. The numerical experiments on various benchmark problems for one and two dimensions indicate that our semi-implicit method obtains well resolved shock, rarefaction and contact solutions. Since our method is conservative, we also obtain correct shock speeds. The smooth flow example illustrates the ability of our method to take significantly large time steps for low Mach number flows as compared to explicit methods. In future work we plan to extend our approach to handle two-way coupling between compressible and incompressible flows, as well as fully implicit solid–fluid coupling.

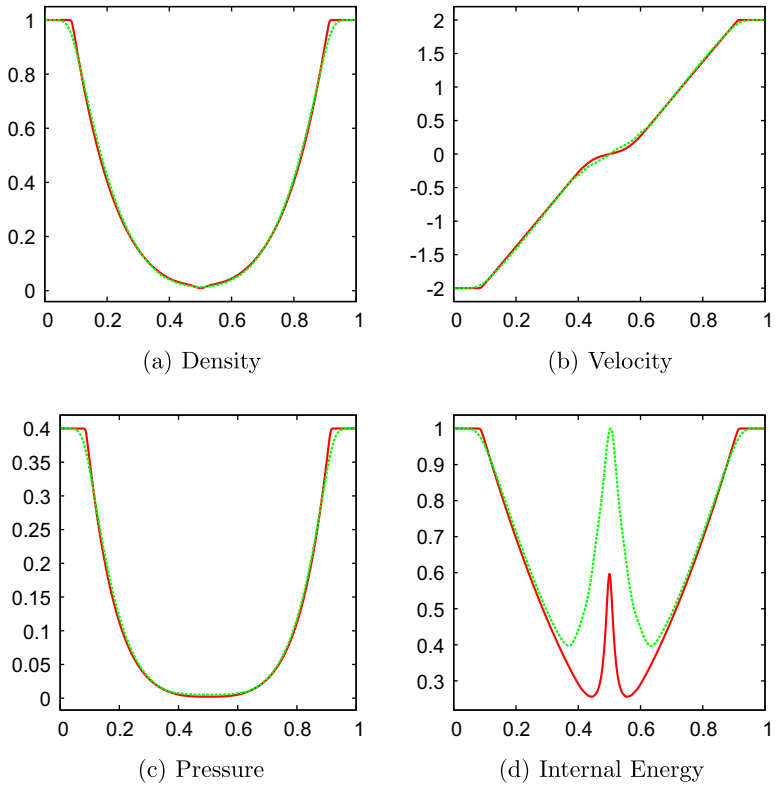


Fig. 11. Numerical results of the symmetric rarefaction shock tube problem at $t = .15$ s. The explicit baseline solution is plotted in red, and the solution from our method is plotted in dotted green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

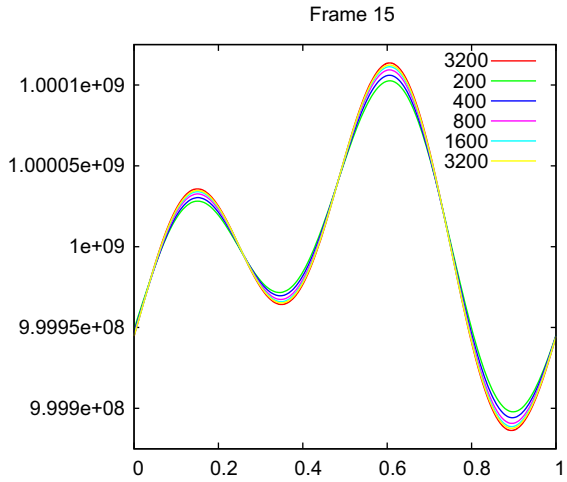


Fig. 12. Numerical results comparing the pressure in smooth flow test at 200, 400, 800, 1600 and 3200 grid cells with an effective sound speed based CFL number 3 at $t = 1.5 \times 10^{-5}$ s. The red curve is the explicit simulation run at 3200 grid cells with a CFL number .5. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

Table 2
Timing results for smooth flow test, with Δt approximately constant. The wall clock times are shown for simulations till $t = 5 \times 10^{-5}$ s.

Grid Resolution	Effective sound speed CFL	Δt	Wall clock time (Implicit)	Wall clock time (Explicit)
3200	3	5.01e−08	63.41 s	511.67 s
32,000	30	5.01e−08	810.03 s	60498.49 s
320,000	300	5.01e−08	9976.58 s	Impractical

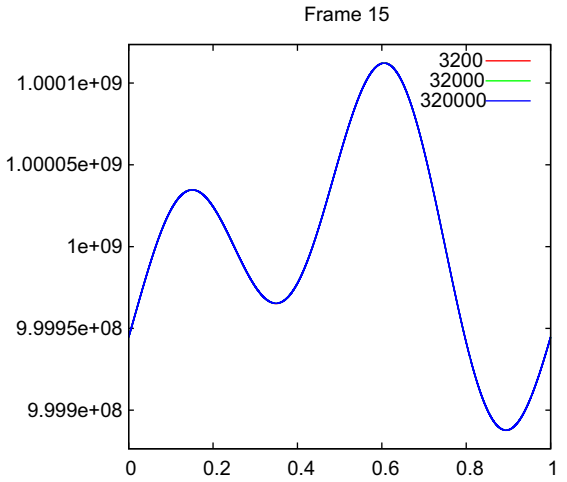


Fig. 13. Numerical results showing pressure in the smooth flow test at 3200, 32,000 and 320,000 grid cells. We used an effective sound speed based CFL number of 3, 30 and 300 respectively at $t = 1.5 \times 10^{-5}$ s. Since Δt stays constant, the solution remains relatively unchanged even as we get huge time step gains.

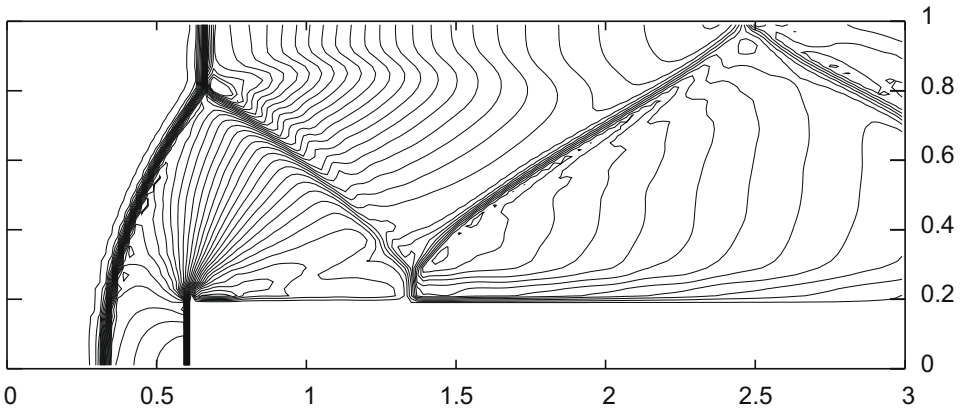


Fig. 14. Numerical results showing the contour plots of density for the flow past a step test on a grid of size 120×40 at $t = 4$ s. Thirty contours are plotted in the range [.2568, 6.067].

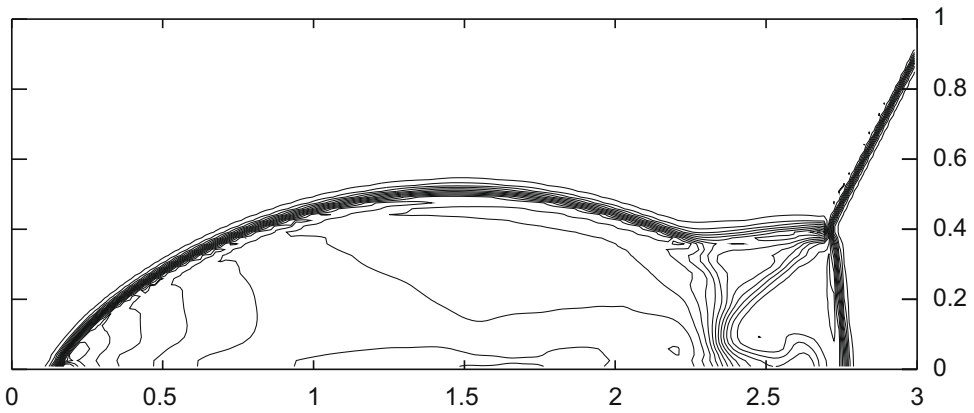
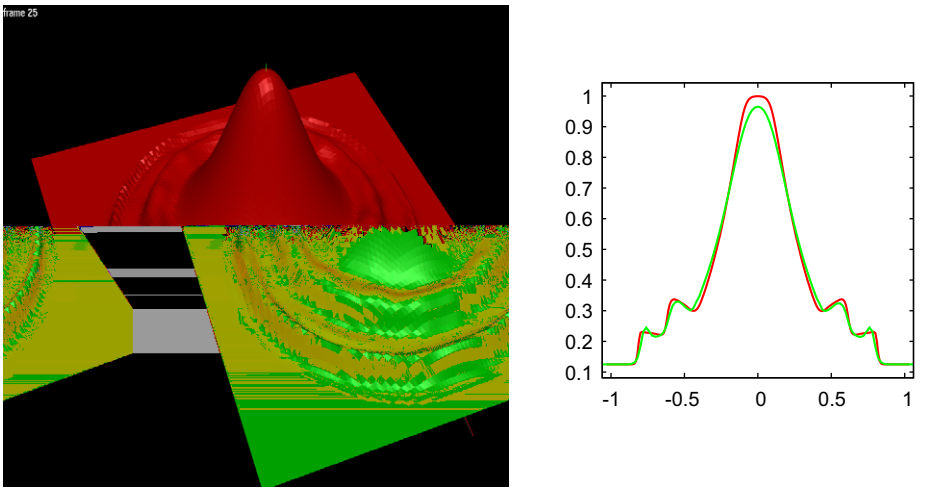


Fig. 15. Numerical results showing the contour plots of density for the double mach reflection of a strong shock on a grid of size 240×60 at $t = .2$ s. Thirty contours are plotted within the range [1.731, 20.92].



(a) Density

(b) 1-D slice of density, with the reference solution shown in red.

Fig. 16. Numerical results for the circular shock test on a grid of size 100×100 at $t = .25$ s.

Acknowledgments

Research supported in part by a Packard Foundation Fellowship, an Okawa Foundation Research Grant, ONR N0014-06-1-0393, ONR N00014-06-1-0505, ONR N00014-02-1-0720, ONR N00014-05-1-0479 for a computing cluster, NIH U54-GM072970, NSF ACI-0323866, NSF IIS-0326388, NSF ITR-0205671 and NSF CCF-0541148. J.S. was supported in part by an NSF Graduate Research Fellowship.

Appendix. Boundary conditions

Fig. 14 requires the handling of inflow and outflow boundary conditions. We define U_{out} to be the outgoing state and U_{in} to be the ingoing state. The outgoing state, U_{out}, is obtained by simple extrapolation whereas the ingoing state, U_{in}, is obtained by attenuating U_{out} towards specified far-field values. After defining U_{out} via extrapolation, we average the primitive variables to cell flux on the boundary of the domain, and use those values to compute a characteristic decomposition. If the *pth* characteristic field indicates ingoing information, then when applying the ENO scheme in this characteristic field we use U_{in} for the ghost node values. Otherwise U_{out} is used. Note for higher order schemes boundary values will be needed for fluxes on the interior of the domain as well, and we choose the ghost nodes (as U_{in} or U_{out}) in the same fashion.

Our ingoing state, U_{in}, is obtained by attenuating the extrapolated state, U_{out}, towards a given far-field state, U_{far}. This is accomplished by multiplying U_{out} with each of the left eigenvectors, attenuating if the eigenvalue in that characteristic field indicates an ingoing wave, and then multiplying by the right eigenvector. Defining the scalar characteristic information in each field as $\xi^p = L^p U_{out}$, we would attenuate ξ^p towards ξ^p_{far} using the analytic solution of the ODE

$$d\xi/dt = K(\xi - \xi_{far})$$

for time step Δt using initial data of $\xi = \xi_{out}$. We used an attenuation coefficient of $K = -.5$ in our examples.

References

[1] R. Courant, E. Issacson, M. Rees, On the solution of nonlinear hyperbolic differential equations by finite differences, Comm. Pure Appl. Math. 5 (1952) 243–255.
[2] R. Fedkiw, X.-D. Liu, S. Osher, A general technique for eliminating spurious oscillations in conservative schemes for multiphase and multispecies Euler equations, Int. J. Nonlinear Sci. Numer. Sim. 3 (2002) 99–106.
[3] R. Fedkiw, A. Marquina, B. Merriman, An isobaric fix for the overheating problem in multimaterial compressible flows, J. Comput. Phys. 148 (1999) 545–578.
[4] R. Fedkiw, B. Merriman, S. Osher, Efficient characteristic projection in upwind difference schemes for hyperbolic systems (the complementary projection method), J. Comput. Phys. 141 (1998) 22–36.
[5] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[6] S.Y. Kadioglu, M. Sussman, Adaptive solution techniques for simulating underwater explosions and implosions, J. Comput. Phys. 227 (2008) 2083–2104.
[7] S.Y. Kadioglu, M. Sussman, S. Osher, J. Wright, M. Kang, A second order primitive preconditioner for solving all speed multi-phase flows, J. Comput. Phys. 209 (2005) 477–503.
[8] M. Kang, R. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360.

[9] S. Osher, C.-W. Shu, High order essentially non-oscillatory schemes for Hamilton–Jacobi equations, SIAM J. Num. Anal. 28 (1991) 902–921.
[10] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[11] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock capturing schemes II (two), J. Comput. Phys. 83 (1989) 32–78.
[12] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (April) (1984) 115–173.
[13] F. Xiao, Unified formulation for compressible and incompressible flows by using multi-integrated moments I: one-dimensional inviscid compressible flow, J. Comput. Phys. 195 (2004) 629–5654.
[14] F. Xiao, R. Akoh, S. Ii, Unified formulation for compressible and incompressible flows by using multi-integrated moments II: multi-dimensional version for compressible and incompressible flows, J. Comput. Phys. 213 (2006) 31–56.
[15] T. Yabe, P.Y. Wang, Unified numerical procedure for compressible and incompressible fluid, J. Phys. Soc. Japan 60 (July) (1991) 2105–2108.
[16] S.Y. Yoon, T. Yabe, The unified simulation for incompressible and compressible flow by the predictor–corrector scheme based on the CIP method, Comput. Phys. Commun. 119 (1999) 149–158.