THE UNIVERSITY OF READING

Interface Tracking using Eulerian Methods with a Lagrangian Phase

P. Sims

Numerical Analysis Report 4/97

DEPARTMENT OF MATHEMATICS

Interface Tracking using Eulerian methods with a Lagrangian phase

Paul Sims

June 1997

Abstract

The author describes a method for solving the equations of gas dynamics for tracking interfaces in multimaterial flow using a Volume of Fluid method. The method firstly involves a Lagrangian phase followed by one of three recovery steps, piecewise constant (1st order), piecewise linear (2nd order) or piecewise parabolic (3rd order). Experiments show that when the recovery steps are applied to the linear advection equation there is a large improvement in the final solution with each increase in accuracy. However, this improvement does not appear when solving the gas dynamics equations when moving from piecewise linear to piecewise parabolic recovery. An error has also been identified when the interface does not align with the diaphragm and consequently the contact discontinuity in the Sod problem.

Acknowledgments

I would like to thank Mike Baines, Paul Glaister (University of Reading, Mathematics Department) and Brian Jones (Atomic Weapons Establishment (AWE), Aldermaston) for their help and guidance on this work.

I would also like to acknowledge the financial support of the AWE.

Contents

Chapter 1

Introduction

The solution of hyperbolic PDE's has numerous applications. However, only very simple cases are solvable analytically. This has meant that the development of accurate and robust numerical schemes has become an important area of research.

One such application is in the study of compressible flow in computational fluid dynamics (CFD). Here shock fronts can develop from relatively smooth initial conditions. It is these discontinuities in the flow that can make numerical schemes give very poor results. First-order schemes like Godunov's Method [7] smear out these shocks while second-order methods like Lax-Wendroff produce spurious oscillations which can make positive variables like density take negative values.

Numerical methods for solving the equations of gas dynamics can be split into groups, finite element, finite difference/volume, spectral methods etc. This paper concentrates on finite difference/volume methods. These methods can be further split into subgroups, Eulerian and Lagrangian. In Eulerian methods the numerical mesh remains in a fixed position and the various flow variables are calculated at specified points on the mesh (usually at either grid nodes or grid centres) at each time step. In Lagrangian methods the mesh moves to follow particular fluid elements as time passes. Both types of method have advantages and disadvantages. Of particular interest in the present investigation is interface tracking in multimaterial flow. Here Eulerian schemes have problems tracking the interface so a Lagrangian scheme may appear to be the obvious solution. In one dimension this may be a satisfactory approach but in two dimensions distortion and tangling of the Lagrangian mesh makes it unacceptable.

Schemes have been developed that capture the advantages of both methods without the disadvantages. These are referred to in papers by Van Leer [11, 12] and Colella and Woodward [3] and describe Eulerian schemes with a Lagrangian phase. The basic steps are identical for schemes of this nature. Firstly the Lagrangian equations are solved for a single timestep and new

values of the particular variables are found along with the new Lagrangian mesh values. Secondly a remap or advection step is performed to move the new Lagrangian values back on to the Eulerian grid.

The schemes in [11, 12, 3] are slightly different from the ones presented in this paper. Here the Lagrangian phase is calculated using a method given by Debar [5] using artificial viscosities. The remap steps are similar to those described in [11], using a piecewise constant or piecewise linear reconstruction and in [3] a piecewise parabolic reconstruction.

Two test cases have been used, Sod's shock problem and the Blast Wave problem (see [15]), to determine the robustness and accuracy of the methods.

To summarise this paper: Chapter 2 firstly introduces three advection schemes and applies them to the linear advection equation in one dimension. These three schemes are then applied, along with a Lagrangian phase, to the gas dynamics equations as recovery or remap steps. This chapter provides a summary of the work done by Stokes [14]. Chapter 3 introduces the changes needed to be made to the schemes so that they will track interfaces between different materials. Finally Chapter 4 provides a conclusion for this work.

Chapter 2

Background

2.1 Numerical Schemes

The methods described here are applied to the linear advection equation but they can easily be translated to apply to the gas dynamics equations. The basic equation is

$$\frac{\partial a}{\partial t} + \lambda \frac{\partial a}{\partial x} = 0 \tag{2.1}$$

where λ is the advection velocity.

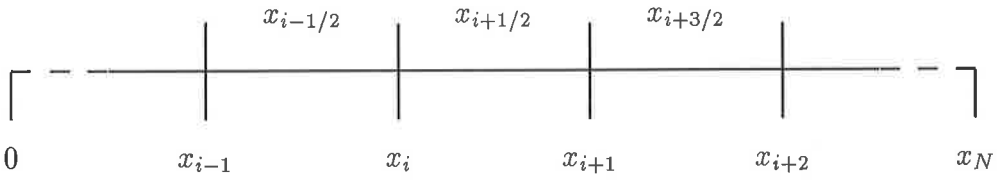

Figure 2.1: Numerical grid.

The notation in the methods described below refers to the, not necessarily uniform, grid in Figure 2.1 with the $i+1/2$'th cell situated between the nodes at x_i and x_{i+1}. The cell volume is given by

$$\Delta x_{i+1/2} = x_{i+1} - x_i \tag{2.2}$$

and we define

$$\Delta x_i = \frac{\Delta x_{i+1/2} + \Delta x_{i-1/2}}{2} \tag{2.3}$$

The initial discretisation is similar for all three methods. The steps are as follows:

1. Given the initial function $a(t_0, x)$, determine the mesh averages

$$a_{i+1/2} = \frac{1}{\Delta x_{i+1/2}} \int_{x_i}^{x_{i+1}} a(t_0, x) dx \tag{2.4}$$

2. Replace the original distribution by either a piecewise constant, linear or parabolic function.

3. Integrate over a finite time step Δt subject to the usual CFL condition

$$\sigma \leq 1 \tag{2.5}$$

4. Determine the new mesh averages.

2.1.1 Piecewise Constant Method

Here the initial distribution is replaced by a piecewise constant distribution as in Godunov's method and is advected some finite distance $\lambda \Delta t$, see Figure 2.2.

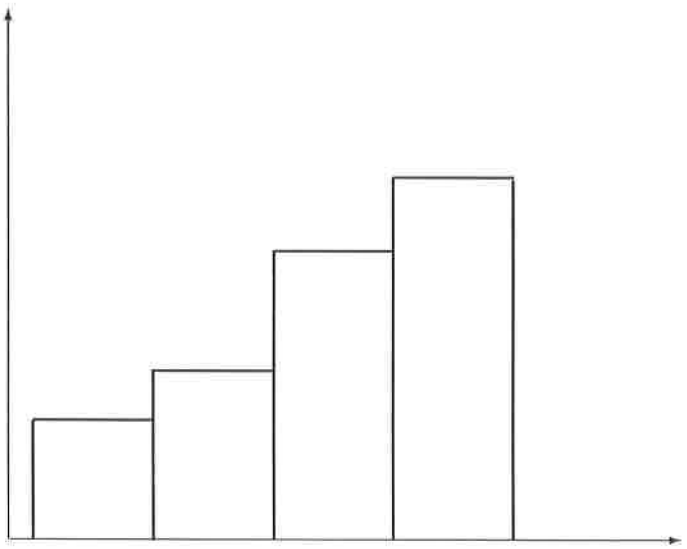

Figure 2.2: Advection of the piecewise constant distribution

The updated average in cell $i + 1/2$ can be calculated using the conservative formula

$$a_{i+1/2}^{n+1} = a_{i+1/2}^{n} + \frac{\Delta t}{\Delta x_{i+1/2}}(f_i - f_{i+1}) \tag{2.6}$$

where the intercell numerical fluxes f_i for this method are given by

$$f_i = \begin{cases} \lambda a_{i-1/2} & \text{if } \lambda \geq 0 \\ \lambda a_{i+1/2} & \text{if } \lambda < 0 \end{cases} \tag{2.7}$$

This is simply the CIR or first order upwind scheme applied to the linear advection equation.

2.1.2 Piecewise Linear Method

This method, developed by Van Leer [11], approximates the initial value function by a piecewise linear distribution (see Figure 2.3).

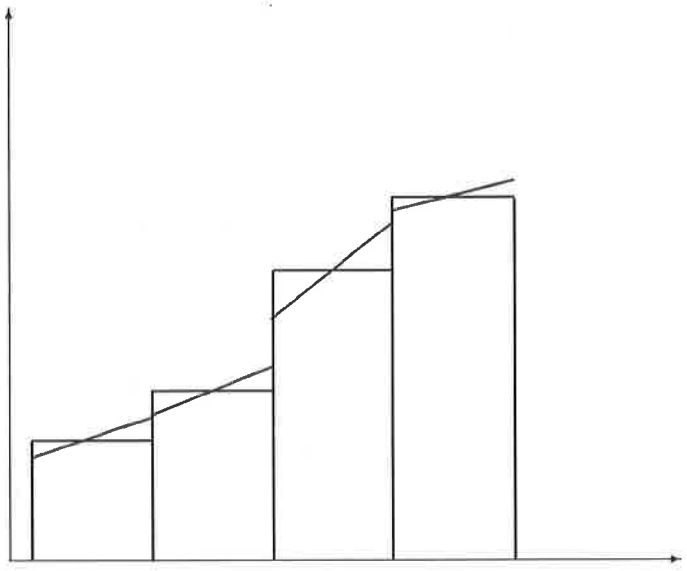

Figure 2.3: Piecewise linear approximation.

As in the piecewise constant scheme the updated average for cell $i + 1/2$ is calculated using (2.6). The difference occurs in the way that the fluxes f_i are defined. In the piecewise constant case $\Delta t f_i$ defined a rectangular area which crosses the intercell boundary. Here these areas are trapezoids and so the calculation is only slightly more complicated.

$$f_i = \begin{cases} \lambda \left[a_{i-1/2} + \frac{\Delta a_{i-1/2}}{2\Delta x_{i-1/2}} (\Delta x_{i-1/2} - \lambda \Delta t) \right] & \text{if } \lambda \geq 0 \\ \lambda \left[a_{i+1/2} - \frac{\Delta a_{i+1/2}}{2\Delta x_{i+1/2}} (\Delta x_{i+1/2} + \lambda \Delta t) \right] & \text{if } \lambda < 0 \end{cases} \tag{2.8}$$

All that is left to do here is to define the slope of the line

$$\frac{\partial a}{\partial x} \approx \frac{\Delta a_{i+1/2}}{\Delta x_{i+1/2}} \tag{2.9}$$

Scheme 1 from [11] is used here. This defines Δa by centrally differencing a, i.e.

$$\Delta a_{i+1/2} = \frac{1}{2}(a_{i+3/2} - a_{i-1/2}) \tag{2.10}$$

As the scheme then stands it is 2nd order accurate but may produce spurious oscillations when used. However, it can be made completely monotonic, in the linear advection case, by applying a monotonicity algorithm given in [11] so that the slope is now given by

$$(\Delta a_{i+1/2})_{mono} = \begin{cases} \alpha sgn(a_{i+3/2} - a_{i-1/2}) & \text{if } \beta > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.11}$$

where

$$\alpha = min(\frac{1}{2}|a_{i+3/2} - a_{i-1/2}|, 2|a_{i+3/2} - a_{i+1/2}|, 2|a_{i+1/2} - a_{i-1/2}|)$$

$$\beta = (a_{i+3/2} - a_{i+1/2})(a_{i+1/2} - a_{i-1/2})$$

2.1.3 Piecewise Parabolic Method (PPM)

This method, developed by Colella and Woodward [3], approximates the initial value function by a piecewise continuous parabolic distribution (see Figure 2.4).

As in the piecewise constant and linear schemes the updated average for the $i + 1/2$'th cell can be calculated using (2.6). This scheme is much more complicated than the two previous ones and we simply describe a summary of what is given in [3]. In this sub-section the subscript of cell centre $i + 1/2$ will be replaced by j as the equations are quite complicated and this makes for easier reading.

Let a be given by a parabolic profile in each zone, i.e.

$$a(x) = a_{L,j} + z(\Delta a_j + a_{6,j}(1 - z)) \tag{2.12}$$

where

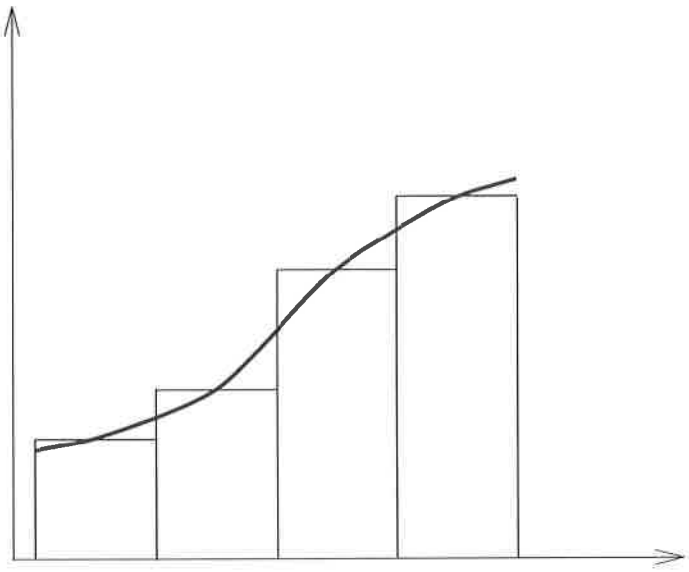

Figure 2.4: Piecewise Parabolic approximation.

$$z = \frac{x - x_i}{\Delta x_j}, \qquad x_i \leq x \leq x_{i+1} \tag{2.13}$$

with

$$\Delta a_j = a_{R,j} - a_{L,j} \tag{2.14}$$

$$a_{6,j} = 6(a_j^n - \frac{1}{2}(a_{L,j} + a_{R,j})) \tag{2.15}$$

We calculate $a_{L,j}$ and $a_{R,j}$ first by interpolating to calculate a_i, an approximation of a at the node x_i.

$$\begin{aligned} a_{i+1} \;=\; & a_j + \frac{\Delta x_j}{\Delta x_j + \Delta x_{j+1}}(a_{j+1} - a_j) + \frac{1}{\sum_{k=-1}^{2} \Delta x_{j+k}} \times \\ & \left\{ \frac{2\Delta x_{j+1}\Delta x_j}{\Delta x_j + \Delta x_{j+1}} \left[\frac{\Delta x_{j-1} + \Delta x_j}{2\Delta x_j + \Delta x_{j+1}} - \frac{\Delta x_{j+2} + \Delta x_{j+1}}{2\Delta x_{j+1} + \Delta x_j} \right] (a_{j+1} - a_j) \right. \\ & \left. - \Delta x_j \frac{\Delta x_{j-1} + \Delta x_j}{2\Delta x_j + \Delta x_{j+1}} \delta_m a_{j+1} + \Delta x_{j+1} \frac{\Delta x_{j+1} + \Delta x_{j+2}}{\Delta x_j + 2\Delta x_{j+1}} \delta_m a_j \right\} \end{aligned} \tag{2.16}$$

with

$$\delta_m a_j = \begin{cases} \min(|\delta a_j|, 2|a_j - a_{j-1}|, 2|a_{j+1} - a_j|)\mathrm{sgn}(\delta a_j), \\ \qquad\qquad \text{if } (a_{j+1} - a_j)(a_j - a_{j-1}) > 0 \\ 0 \qquad\qquad \text{otherwise} \end{cases} \tag{2.17}$$

where

$$\begin{aligned} \delta a_j &= \frac{\Delta x_j}{\Delta x_{j-1} + \Delta x_j + \Delta x_{j+1}} \\ &\quad \times \left[\frac{2\Delta x_{j-1} + \Delta x_j}{\Delta x_{j+1} + \Delta x_j}(a_{j+1} - a_j) + \frac{\Delta x_j + 2\Delta x_{j+1}}{\Delta x_{j-1} + \Delta x_j}(a_j - a_{j-1}) \right] \end{aligned}$$

This equation helps to improve the representation of discontinuities. With these definitions $a_{L,j}$ and $a_{R,j}$ are defined as

$$a_{L,j} = a_{R,j-1} = a_i \tag{2.18}$$

This scheme is now 3rd order accurate but may again produce spurious oscillations unless a monotonicity algorithm is applied. For most of the time $a_{L,j}$ and $a_{R,j}$ will be assigned as in (2.18). However if a_j is a local maximum or minimum then the interpolation function is set to be a constant (2.19). If either $a_{L,j}$ or $a_{R,j}$ is sufficiently close to a_j the interpolation parabola takes on values which are not between $a_{L,j}$ and $a_{R,j}$. If this is the case either (2.20) or (2.21) is applied. Here $a_{L,j}$ or $a_{R,j}$ is reset such that the opposite edge of the cell from the one being reset has a zero derivative. In summary,

$$a_{L,j} \to a_j, \quad a_{R,j} \to a_j \quad \text{if} \quad (a_{R,j} - a_j)(a_j - a_{L,j}) \leq 0 \tag{2.19}$$

$$\begin{aligned} a_{L,j} &\to 3a_j - 2a_{R,j} \\ &\quad \text{if} \quad (a_{R,j} - a_{L,j})\left(a_j - \frac{a_{L,j} + a_{R,j}}{2}\right) > \frac{(a_{R,j} - a_{L,j})^2}{6} \end{aligned} \tag{2.20}$$

$$\begin{aligned} a_{R,j} &\to 3a_j - 2a_{L,j} \\ &\quad \text{if} \quad -(a_{R,j} - a_{L,j})\left(a_j - \frac{a_{L,j} + a_{R,j}}{2}\right) > \frac{(a_{R,j} - a_{L,j})^2}{6} \end{aligned} \tag{2.21}$$

Once $a_{L,j}$ and $a_{R,j}$ have been calculated we can write down an explicit expression for a_j^{n+1} given by

$$\begin{array}{rcl} h_{i,L}^a(y) & = & \frac{1}{y}\int_{x_{i-y}}^{x_i} a(x)dx \\ h_{i,R}^a(y) & = & \frac{1}{y}\int_{x_i}^{x_{i+y}} a(x)dx \end{array} \tag{2.22}$$

Assuming z is positive this results in:

$$\begin{array}{rcll} h_{i,L}^a(y) & = & a_{R,j-1} - \frac{z}{2}\left(\Delta a_{j-1} - \left(1 - \frac{2}{3}z\right)a_{6,j-1}\right) & \text{for } z = \frac{y}{\Delta x_{j-1}} \\ h_{i,R}^a(y) & = & a_{L,j} - \frac{z}{2}\left(\Delta a_j - \left(1 - \frac{2}{3}z\right)a_{6,j}\right) & \text{for } z = \frac{y}{\Delta x_j} \end{array} \tag{2.23}$$

Then these can be substituted into the explicit conservative formula

$$a_j^{n+1} = a_j^n + \frac{\Delta t}{\Delta x_j}(f_i - f_{i+1}) \tag{2.24}$$

where

$$f_i = \begin{cases} \lambda h_{i,L}^a(\lambda\Delta t), & \text{if } \lambda \geq 0 \\ \lambda h_{i,R}^a(-\lambda\Delta t), & \text{if } \lambda < 0 \end{cases} \tag{2.25}$$

This completes the descriptions of the advection algorithms used for the linear advection equation. They differ only very slightly for the gas dynamics case as will be pointed out later.

These interpolation methods could be made even higher order accurate by using higher order polynomials such as cubics. A piecewise monotone cubic interpolation method is given by Fritsch and Carlson [6] but will not be discussed here.

2.1.4 Numerical Experiments

A comparison is made here for the three different schemes when a square wave is advected through the mesh with 100 space intervals and CFL condition of 0.5. In Figure 2.5 the square wave has been advected through the mesh once and in Figure 2.6 it has been advected through the mesh 10 times. Periodic boundary conditions apply in all cases.

These graphs clearly show that the first order method is of little use here. The quadratic reconstruction produces the best results, as expected since it is a 3rd order method. This inevitably increases the running time of the code. The results for the quadratic reconstruction after advection through the mesh 10 times are comparable with those of the linear reconstruction after once through the mesh. This implies that the increased accuracy of the PPM scheme outweighs the decreased running time of the piecewise linear method. Table 2.1 gives a measure of the accuracy and convergence of the 3

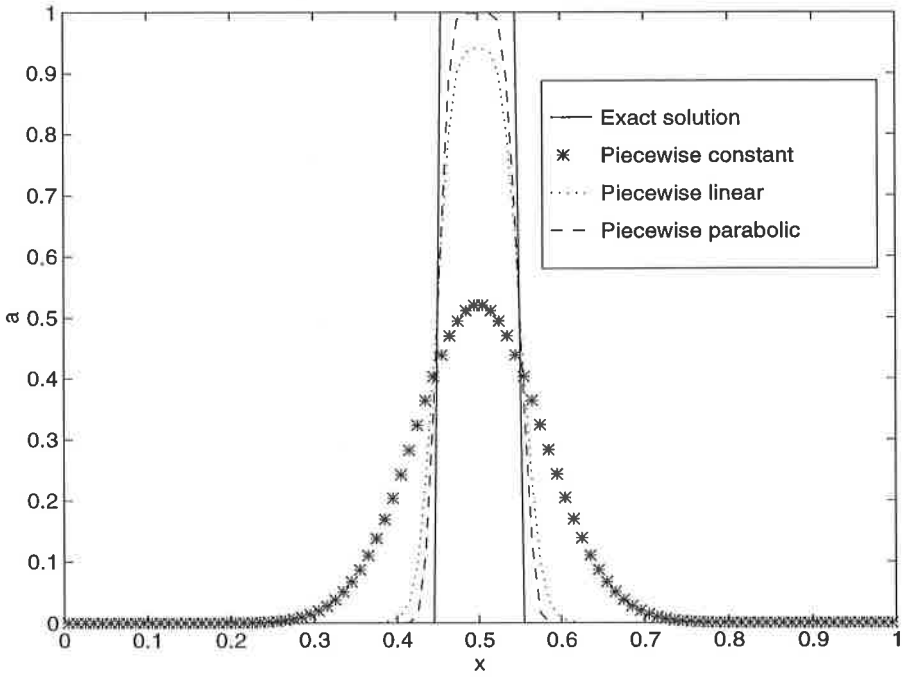

Figure 2.5: Square wave advected through mesh once. 100 cells, CFL=0.5

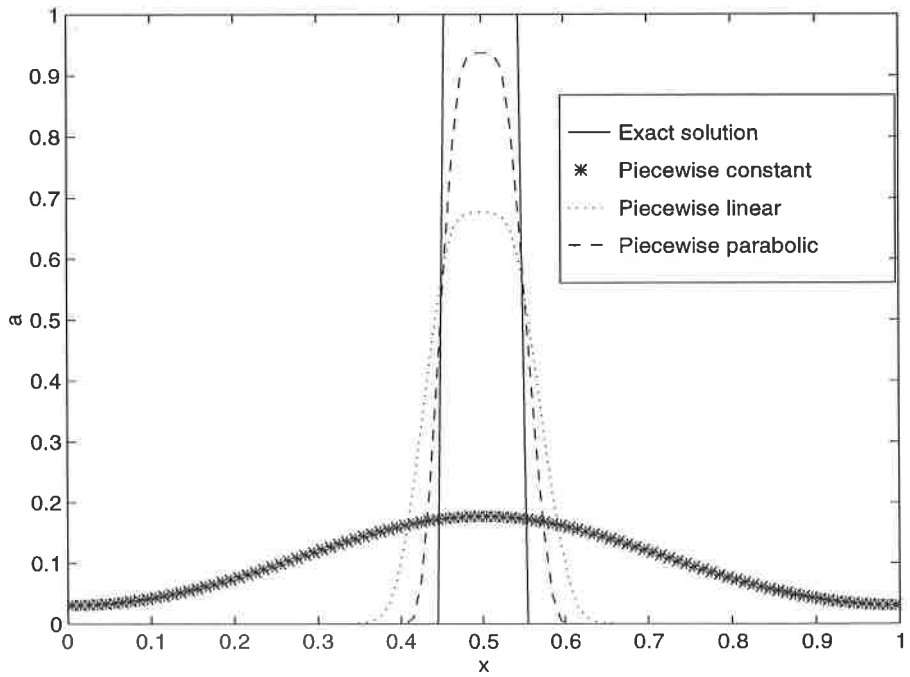

Figure 2.6: Square wave advected through mesh 10 times. 100 cells, CFL=0.5

No. of cells	Constant	Linear	Parabolic
100	0.102733	0.031923	0.019397
200	0.078056	0.018196	0.011134
400	0.056305	0.010637	0.006410
800	0.039887	0.006223	0.003708

Table 2.1: Convergence rates for scalar advection of a square wave advected through the mesh once using the 3 different advection steps. Measured error is the average value of $|(a - a_{exact})|$.

schemes with the 3rd order PPM scheme giving the best results as observed in Figures 2.5 and 2.6. All the schemes appear to be completely monotonic for the linear advection equation. In the next section these schemes are applied the equation of gas dynamics.

2.2 Gas Dynamics Equation

All the schemes mentioned here for solving the equation of gas dynamics use a two step method. The first step is a Lagrangian phase in which the grid moves with the fluid. New (temporary) grid point positions are calculated here which can be used along with the original grid points to calculate volume fluxes which can be passed on to the second step. This step was implicit in the methods described in section 2.1. The second step is called an advection, remap or recovery step. Here the methods described in section 2.1 are used to remap the values from the Lagrangian phase back through the fluid onto the original Eulerian grid. Here $\lambda\Delta t$ is replaced by the volume fluxes calculated in the Lagrangian phase.

2.2.1 Lagrangian phase

Here we solve the Lagrangian equations of gas dynamics. The standard equation for one dimension are given and can be found in [4].

Conservation of Mass

$$(\rho x_m)_t = 0 \tag{2.26}$$

Conservation of Momentum

$$\rho \frac{\partial u}{\partial t} = -\frac{\partial p}{\partial x} \tag{2.27}$$

Conservation of Energy

$$\frac{\partial \varepsilon}{\partial t} = -\frac{p}{\rho}\frac{\partial p}{\partial x} \tag{2.28}$$

The Equation of State for an Ideal Gas is taken to be

$$p = (\gamma - 1)\rho\varepsilon \tag{2.29}$$

When discretising these equations, pressure (p), internal energy (ε) and density (ρ) are stored as cell averages at the cell centres. These quantities are denoted by subscript '$i+1/2$'. The velocity (u) is stored at the cell nodes and is given subscript 'i'. The grid is shown in Figure 2.1. The discretised equations of (2.30-2.36) are:

Momentum

$$u_i^{n+1} = u_i^n - \frac{\Delta t[p_{i+1/2}^{n+\frac{1}{2}} + q_{i+1/2}^n - p_{i-1/2}^{n+\frac{1}{2}} - q_{i-1/2}^n]}{M_i^n} \tag{2.30}$$

$$M_i = \frac{1}{2}(M_{i-1/2} + M_{i+1/2}) \tag{2.31}$$

$$M_{i+1/2} = \Delta x_{i+1/2}\rho_{i+1/2} \tag{2.32}$$

Node motion

$$x_i^{n+1} = x_i^n + \frac{1}{2}\Delta t(u_i^n + u_i^{n+1}) \tag{2.33}$$

Mass

$$\rho_{i+1/2}^{n+1} = \frac{\Delta x_{i+1/2}^n \rho_{i+1/2}^n}{\Delta x_{i+1/2}^{n+1}} \tag{2.34}$$

Energy

$$\varepsilon_{i+1/2}^{n+1} = \varepsilon_{i+1/2}^n - \frac{\Delta t}{2}\frac{[u_{i+1}^n + u_{i+1}^{n+1} - u_i^n - u_i^{n+1}](p_{i+1/2}^{n+\frac{1}{2}} + q_{i+1/2}^n)}{M_{i+1/2}^n} \tag{2.35}$$

Equation of State gives

$$p_{i+1/2}^{n+1} = (\gamma - 1)\rho_{i+1/2}^{n+1}\varepsilon_{i+1/2}^{n+1} \tag{2.36}$$

where

$$p_{i+1/2}^{n+\frac{1}{2}} = \frac{1}{2}(p_{i+1/2}^n + p_{i+1/2}^{n+1}) \tag{2.37}$$

This is now an implicit set of difference equations which can be solved by a predictor-corrector method. For the predictor step we take $p^{n+\frac{1}{2}}_{i+1/2} = p^n_{i+1/2}$ for each particular variable and for the corrector step we use (2.37). The correction step is only required once for formal second order accuracy.

One quantity yet to be defined is q. This is known as the artificial viscous pressure or artificial viscosity and is included in the equations to decrease oscillations in the numerical solution. Artificial viscosity is, in general, problem dependent. Two artificial viscosities are given here: the HEMP artificial viscosity and the Christensen flux-limited viscosity. Both viscosities use the same equation for q but define Δu differently (see [1]).

$$q = \begin{cases} c_0 \rho_{i+1/2} (\Delta u)^2 + c_L c_{i+1/2} |\Delta u| & \text{if } \Delta u < 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.38}$$

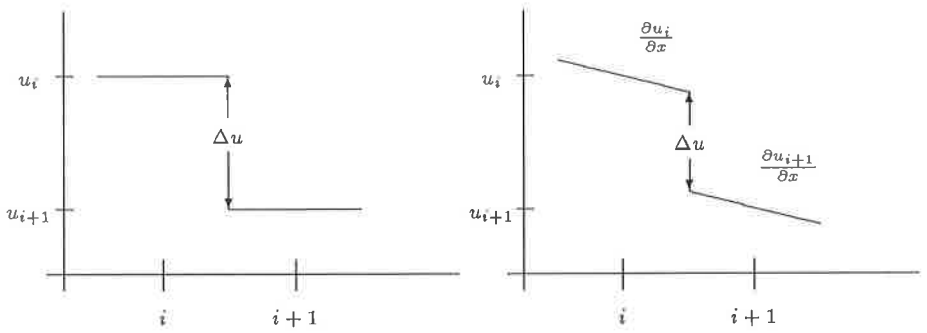

Figure 2.7: The Hemp artificial viscosity approximates the velocity field as a piecewise constant function. The Christensen flux-limited viscosity uses a first order Taylor expansion to improve the accuracy of the velocity jump. The slopes are limited by using Van Leer's monotonic MUSCL difference scheme.

As is shown in Figure 2.7, Δu for the HEMP artificial viscosity is simply given by

$$\Delta u = u_{i+1} - u_i \tag{2.39}$$

In the Christensen flux-limited viscosity Δu is given by using a Taylor expansion of the velocities at the nodes and keeping only the first order terms, so that

$$\Delta u = \left(u_{i+1} - \frac{\Delta X}{2} \frac{\partial u_{i+1}}{\partial x} \right) - \left(u_i + \frac{\Delta X}{2} \frac{\partial u_i}{\partial x} \right) \tag{2.40}$$

where ΔX is the length of the element. To get the improved approximation two first order slopes, S_L and S_R, and a second order approximation of the slope S_C are defined as

$$S_L = \frac{2(u_i - u_{i-1})}{x_i - x_{i-1}}, \quad S_R = \frac{2(u_{i+1} - u_i)}{x_{i+1} - x_i} \tag{2.41}$$

$$S_C = \frac{\left(\frac{x_{i+1}-x_i}{x_i-x_{i-1}}\right)(u_i - u_{i-1}) + \left(\frac{x_i-x_{i-1}}{x_{i+1}-x_i}\right)(u_{i+1} - u_i)}{x_{i+1} - x_{i-1}} \tag{2.42}$$

Monotonicity determines which of these slopes are used and $\frac{\partial u_i}{\partial x}$ is given by

$$\frac{\partial u_i}{\partial x} = \frac{1}{2}[sgn(S_L) + sgn(S_R)]min(|S_L|, |S_R|, |S_C|) \tag{2.43}$$

The values of the constants c_0 and c_L are problem dependent.

2.2.2 Remap step

In this step we move the mesh back to the Eulerian grid where it started before the Lagrangian phase. The basic remap routine is the same for all three methods so that it can be written as one subroutine of the code. The only difference comes when defining the effective densities, specific energies and velocities. These are calculated similarly as in the linear advection case in (2.7) for the piecewise constant remap, (2.8) for the piecewise linear remap and (2.25) for the piecewise parabolic remap with one slight difference. These equations are the equation of the fluxes, the effective quantities, f_i^* being simply these equations divided by λ. Also where we had $\lambda\Delta t$ it is replaced by δV, the volume flux in the equations. The equations for advecting density, internal energy and velocity are now defined as:

Density

$$\tilde{\rho}_{i+1/2}^{n+1} = \frac{M_{i+1/2} + \delta V_i f_i^* - \delta V_{i+1} f_{i+1}^*}{\Delta x_{i+1/2}} \tag{2.44}$$

where

$$\delta V_i = x_i^{n+1} - x_i^n \tag{2.45}$$

Internal Energy

$$\tilde{\varepsilon}_{i+1/2}^{n+1} = \frac{\varepsilon_{i+1/2}^{n+1} M_{i+1/2}^n + \delta M_i f_i^* - \delta M_{i+1} f_{i+1}^*}{\tilde{M}_{i+1/2}} \tag{2.46}$$

$$\delta M_i = \delta V_i f_i^{\rho} \tag{2.47}$$

where $\tilde{M}_{i+1/2}$ is the post-remap mass, f_i^{ρ} is the effective density and δM_i is the mass flux.

Velocity

Since velocity is stored at the cell nodes and not at cell centres like the other quantities, it needs to be defined separately. One possibility would be to write an extra piece of code that remaps node centred quantities. This would be the best option if the code's speed was a consideration. However it is easier to use a method called Half-Index Shift (HIS) given in papers by Benson [1, 2]. This technique allows the cell centred remap routine to be used by defining two new variables ψ_{j1} and ψ_{j2}:

$$\psi_{i+1/2,1} = u_i \tag{2.48}$$

$$\psi_{i+1/2,2} = u_{i+1} \tag{2.49}$$

These quantities are then remapped in exactly the same way as the internal energy, i.e. just replace ε with $\psi_{i+1/2,1}$ and $\psi_{i+1/2,2}$ in Eq. (2.46) to get the remapped quantities $\tilde{\psi}_{i+1/2,1}$ and $\tilde{\psi}_{i+1/2,2}$. Then these quantities are substituted into (2.50) to find the remapped velocity

$$\tilde{u}_i^{n+1} = \frac{(\tilde{M}_{i-1/2}\tilde{\psi}_{i-1/2,2} + \tilde{M}_{i+1/2}\tilde{\psi}_{i+1/2,1})}{\tilde{M}_{i-1/2} + \tilde{M}_{i+1/2}} \tag{2.50}$$

The last thing to mention is the timestep. The Lagrangian phase requires that a sound wave cannot propagate more than one cell width and the remap step requires that the fluid cannot be advected more than one cell width. The equation to calculate a safe timestep is

$$\Delta t = min(F_c\frac{\Delta x_{i+1/2}}{c_{i+1/2}}, F_v\frac{\Delta x_{i+1/2}}{u_i}, F_v\frac{\Delta x_{i+1/2}}{u_{i+1}}) \tag{2.51}$$

where F_c and F_v are constant which are both set as 0.5 and $c_{i+1/2}$ is the speed of sound in the cell given by

$$c_{i+1/2} = \sqrt{\frac{\gamma p_{i+1/2}}{\rho_{i+1/2}}} \tag{2.52}$$

This completes the description of all three methods for solving the gas dynamics equations.

2.2.3 Numerical Experiments

The first test case is Sod's shock tube problem in which a diaphragm at the centre of the tube ($x = 0.5$) separates two constant states (see Figures 2.8, 2.9 and 2.10). The initial conditions of these states are ρ_K, u_K and p_K

No. of cells	Constant	Linear	Parabolic	Lagrangian
100	0.033293	0.020643	0.019279	0.015537
200	0.020181	0.010871	0.010014	0.008116
400	0.013139	0.006321	0.005825	0.003840
800	0.008648	0.003666	0.003320	0.001989
1600	0.006036	0.002145	0.001898	0.001061

Table 2.2: Convergence in density for the Sod problem using the 3 different recovery steps and the pure Lagrangian code with Hemp artificial viscosity ($c_0 = 1.5$ and $c_L = 0.06$) used in all cases. Measured error is the average value of $|(\rho - \rho_{exact})/\rho_{exact}|$.

($K = L, R$: representing the state on the left and right of the diaphragm respectively).

The results from the Sod problem show an increase in the resolution from the piecewise constant method to the linear reconstruction. However a similar increase in resolution is not seen when moving up to quadratic reconstruction as was seen in the scalar advection case. The only slight difference occurs across the contact discontinuity where in the linear case it is spread over 5 cells and in the parabolic it is only spread over 4. This is probably due to the loss of accuracy from the Lagrangian phase which is only 2nd order accurate and it is this that drags the overall accuracy of the scheme down. Table 2.2 shows clearly that piecewise linear and piecewise parabolic recovery steps produce very similar numerical errors and that the pure Lagrangian calculation appears to produce the best results.

The second test case is a much more severe problem, namely the blast wave problem [15]. Here three constant states are separated by two diaphragms at $x = 0.1$ and $x = 0.9$. The resulting density profiles are given below at eight different output times. See Figures 2.11, 2.12 and 2.13.

The results from the blast wave problem show that the extra effort needed for the quadratic reconstruction is not completely wasted as was implied in the Sod problem. The linear reconstruction tends to give a much larger overshoot in the density profile associated with the smaller pressure jump of the right hand state. A 3rd order accurate Lagrangian phase may be needed to show the full benefits of using quadratic reconstruction. The 1st order piecewise constant method is completely inadequate for this test problem.

The Christensen flux-limited artificial viscosity has been overlooked in the results until now. Figure 2.14 shows a comparison of the Christensen viscosity with the Hemp viscosity for four different cases. (a) and (b) are pure Lagrangian calculations, whilst (c) and (d) include the remap step.

Figure 2.14 (a) and (b) show only small differences and both produce

good results compared to the results given in [15]. Figure 2.14 (c) and (d) compare less favourably with the equivalent times in Figures 2.12 and 2.13. Large overshoots occur in both cases and particularly when piecewise linear recovery is used. Possibly the fact that the Lagrangian phase, the artificial viscosity and the remap step are all 2nd order accurate in this case produces higher dispersion errors.

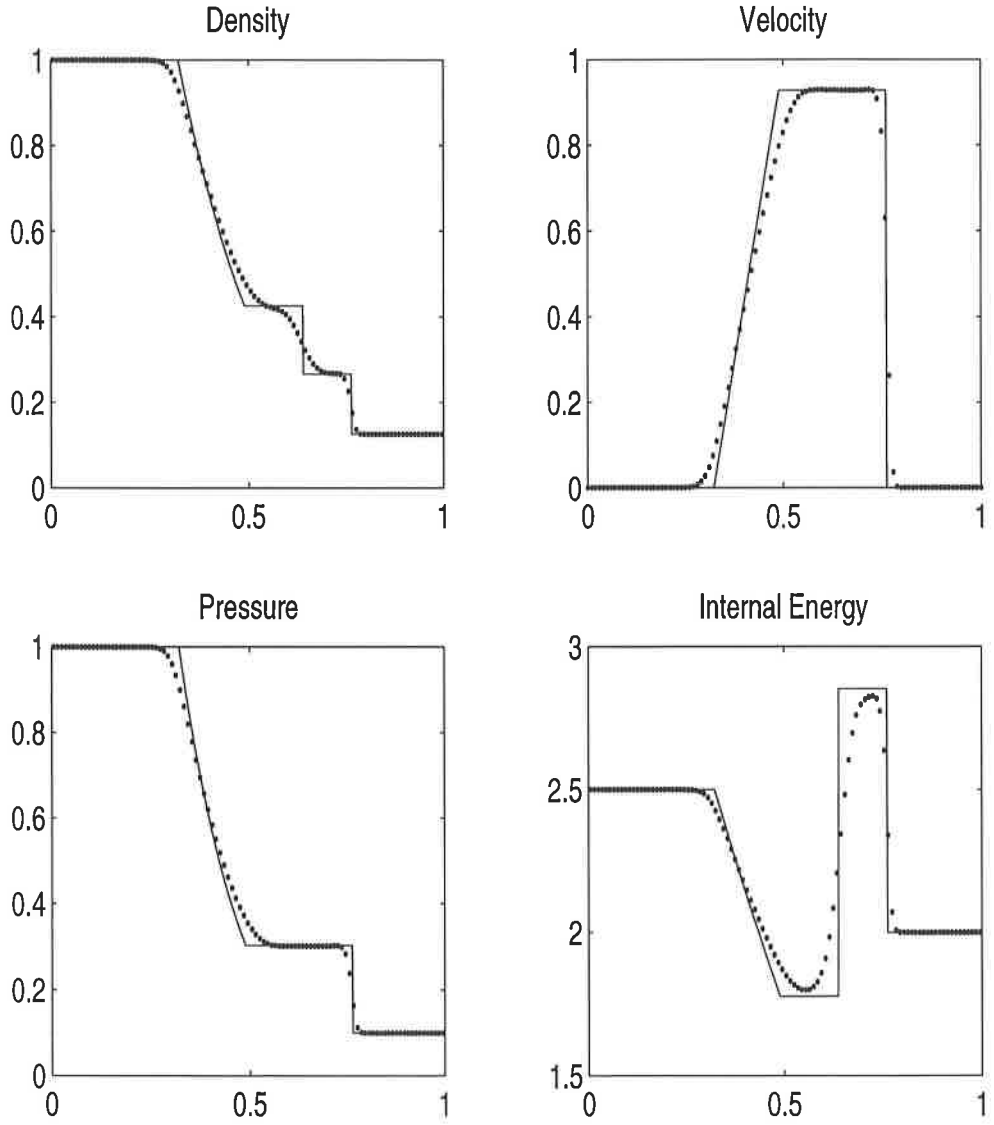

Figure 2.8: Sod's shock tube problem - Plotting exact solution (solid line) and Piecewise constant solution (dotted line). 100 cells. Diaphragm at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$. $\gamma = 1.4$. Output time $t = 0.15$.

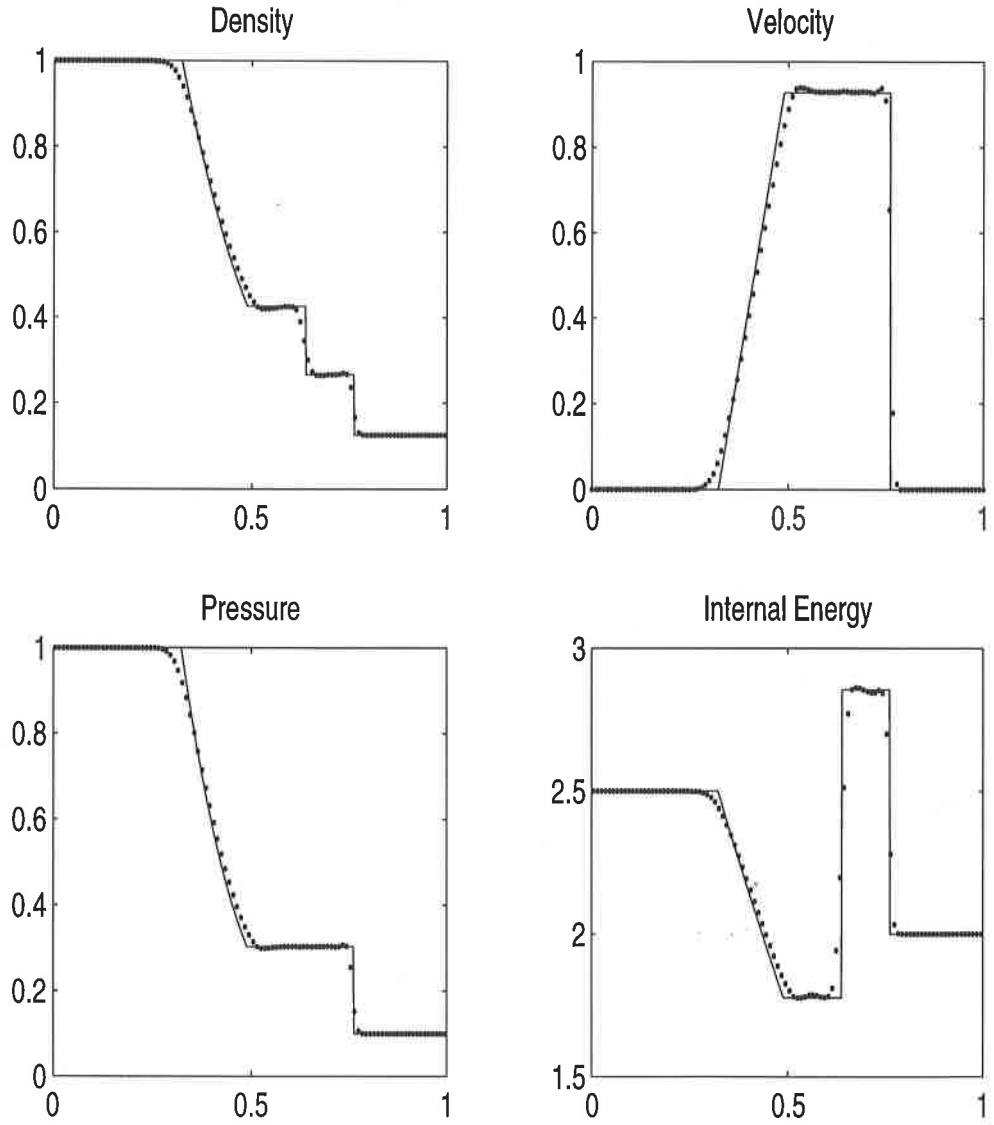

Figure 2.9: Sod's shock tube problem - Plotting exact solution (solid line) and Piecewise linear solution (dotted line). 100 cells. Diaphragm at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$. $\gamma = 1.4$. Output time $t = 0.15$.

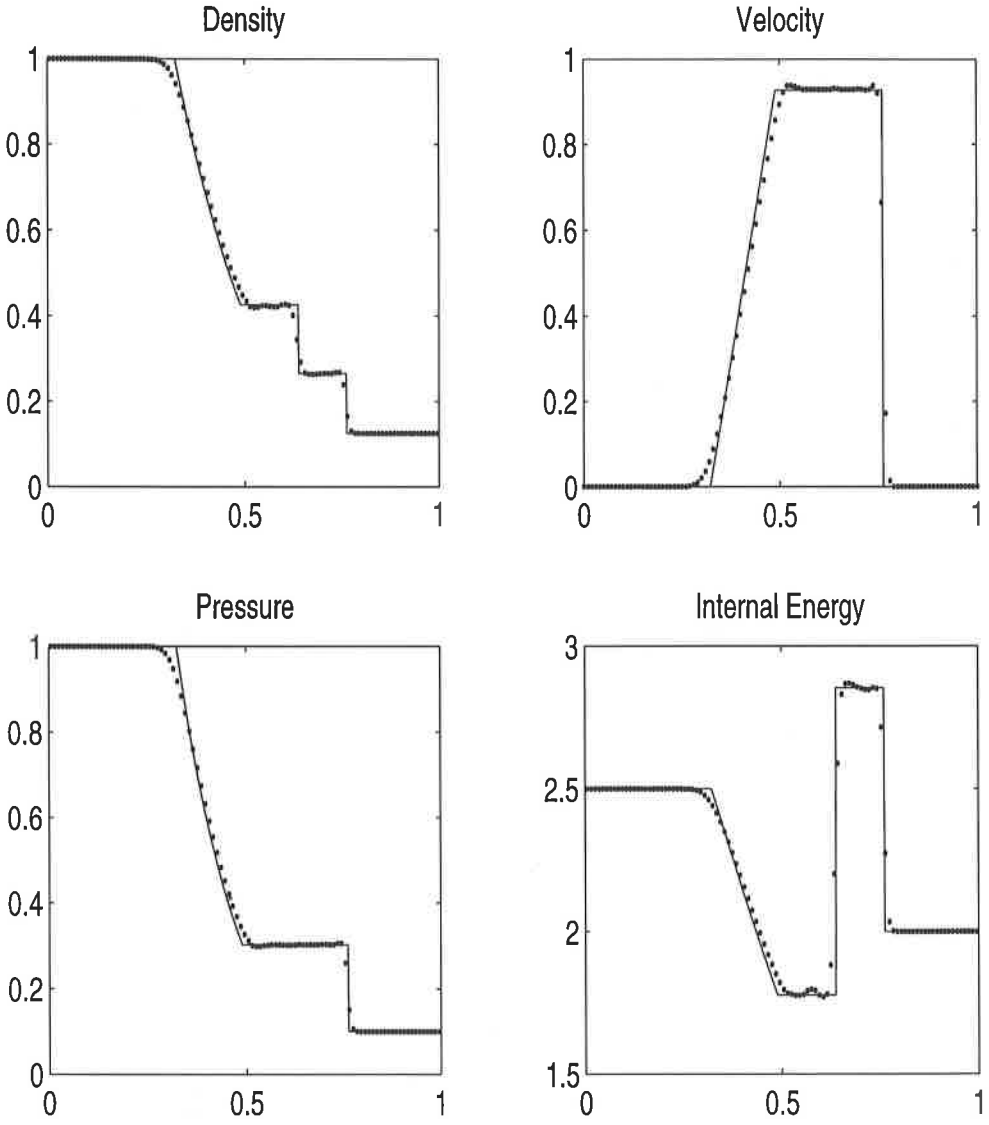

Figure 2.10: Sod's shock tube problem - Plotting exact solution (solid line) and Piecewise parabolic solution (dotted line). 100 cells. Diaphragm at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$. $\gamma = 1.4$. Output time $t = 0.15$.

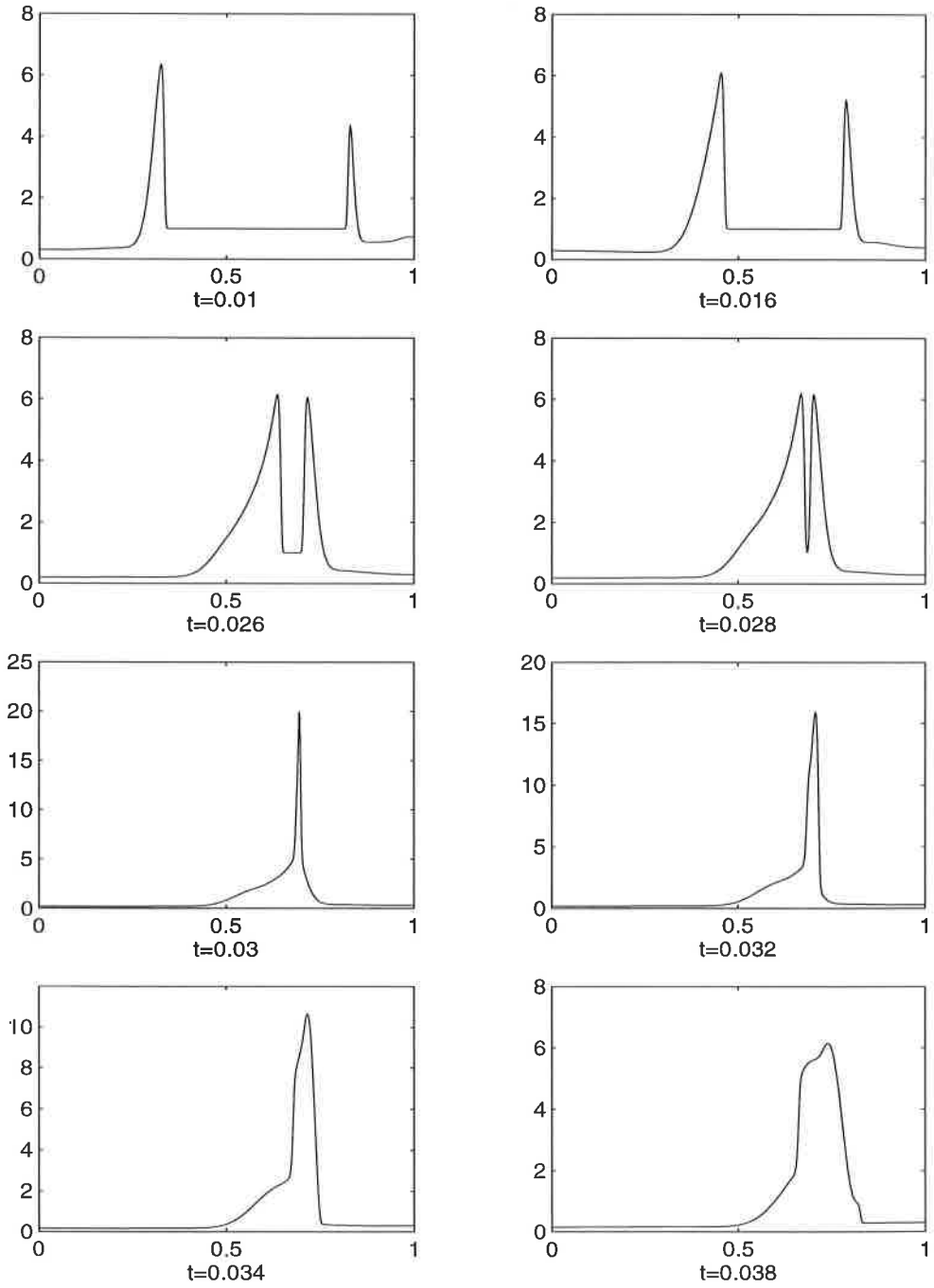

Figure 2.11: Blast Wave problem - Density profile - Piecewise Constant solution. 400 cells. Two diaphragms at 0.1 and 0.9. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1000$. Initial middle state: $\rho_M = 1.0$, $u_M = 0.0$, $p_M = 0.01$. Initial right state: $\rho_R = 1.0$, $u_R = 0.0$, $p_R = 100$. $\gamma = 1.4$.

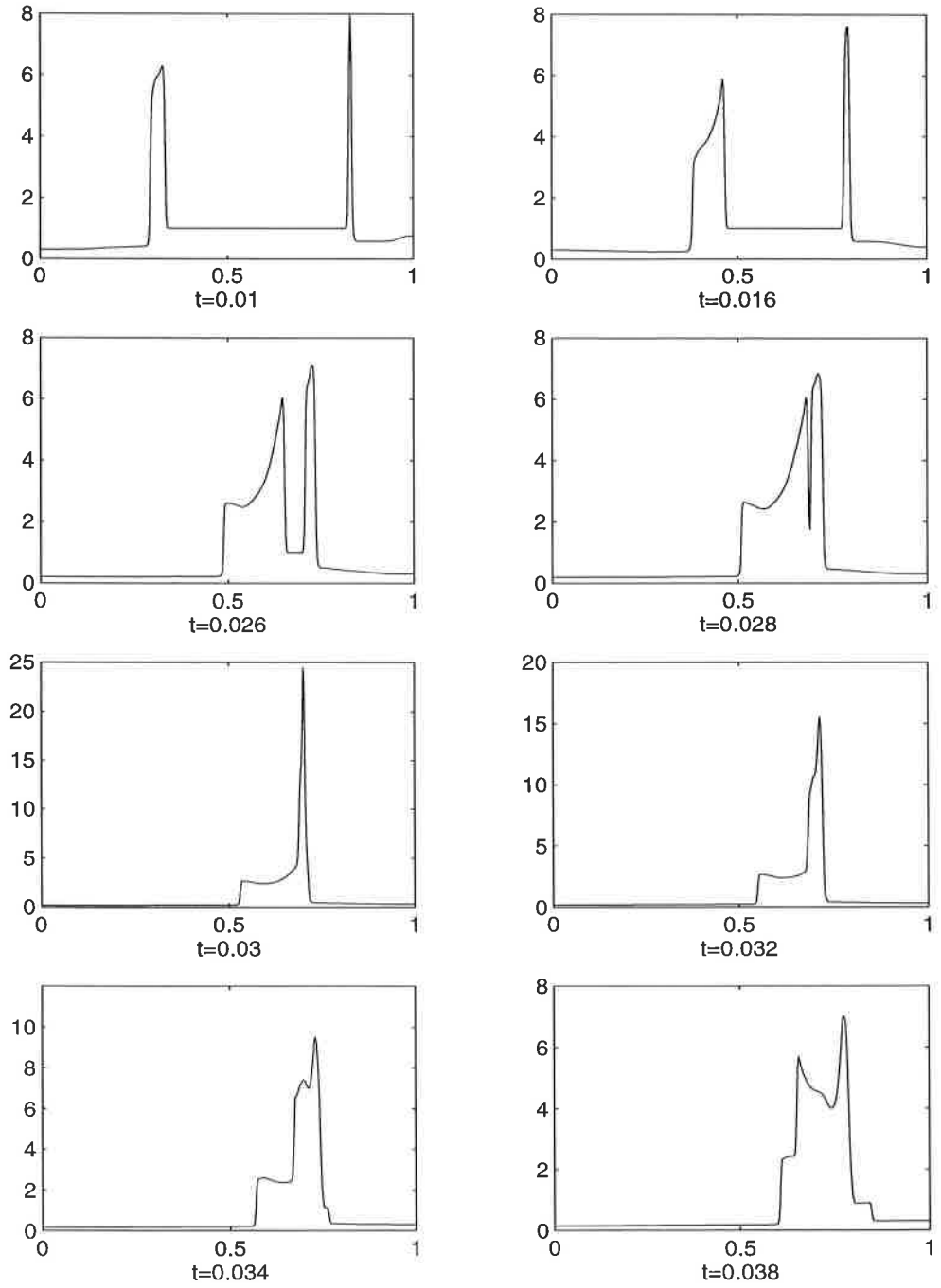

Figure 2.12: Blast Wave problem - Density profile - Piecewise Linear solution. 400 cells. Two diaphragms at 0.1 and 0.9. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1000$. Initial middle state: $\rho_M = 1.0$, $u_M = 0.0$, $p_M = 0.01$. Initial right state: $\rho_R = 1.0$, $u_R = 0.0$, $p_R = 100$. $\gamma = 1.4$.

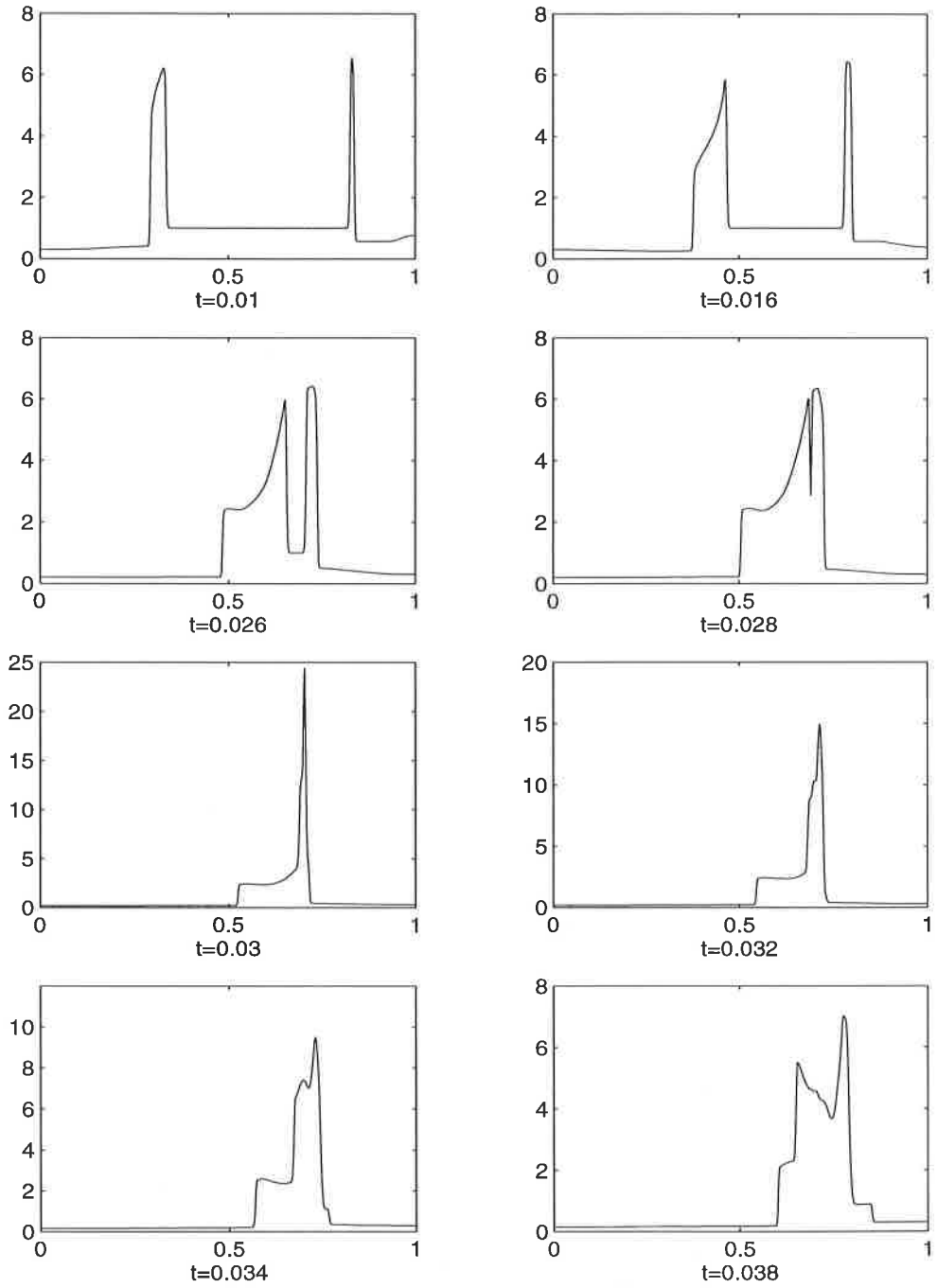

Figure 2.13: Blast Wave problem - Density profile - Piecewise Parabolic solution. 400 cells. Two diaphragms at 0.1 and 0.9. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1000$. Initial middle state: $\rho_M = 1.0$, $u_M = 0.0$, $p_M = 0.01$. Initial right state: $\rho_R = 1.0$, $u_R = 0.0$, $p_R = 100$. $\gamma = 1.4$.

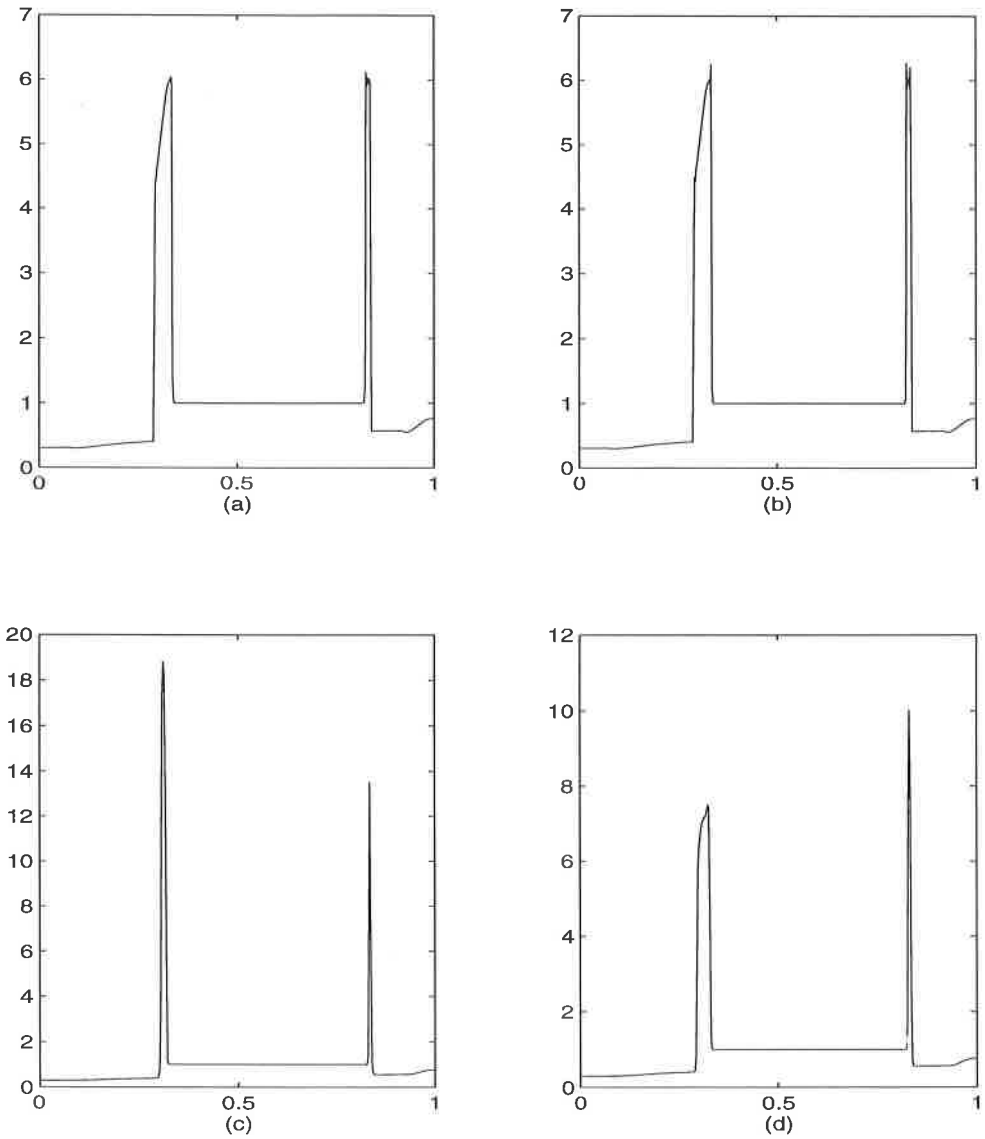

Figure 2.14: Blast Wave problem - Comparison of Artificial Viscosities. 400 cells. (a) Lagrangian code with Hemp viscosity. (b) Lagrangian code with Christensen viscosity. (c) Piecewise linear remap with Christensen viscosity. (d) Piecewise parabolic remap with Christensen viscosity. Two diaphragms at 0.1 and 0.9. Hemp artificial viscosity as coefficients $c_0 = 1.5$ and $c_L = 0.06$. Christensen artificial viscosity as coefficients $c_0 = \frac{2}{3}$ and $c_L = 0.5$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1000$. Initial middle state: $\rho_M = 1.0$, $u_M = 0.0$, $p_M = 0.01$. Initial right state: $\rho_R = 1.0$, $u_R = 0.0$, $p_R = 100$. $\gamma = 1.4$. Output time $t = 0.01$.

Chapter 3

Interface Tracking in 1D

Interface tracking can be done in a number of ways and a summary of three different methods, surface tracking, volume tracking and mesh moving methods, is given by Hyman [9]. Here we shall concentrate only on volume tracking or volume of fluid methods. Various papers have been written that give details of these methods and they include [8, 13, 16, 17]. The volume of fluid method uses fractional volumes of each material in the cell to keep track of the interface position.

The single material methods given in section 2.2 have been applied to multi-fluid flows so that the interface between the different fluids can be tracked. Only two materials have been used here since adding more materials complicates the problem without giving any extra information about the interface tracking procedure.

The procedure used is to go through the methods given above but treating each cell as having two fluids in it. Each cell has a fractional volume associated with each fluid. Provided the cell only has one fluid in it, it can be treated exactly as before. When the cell is mixed the method reduces to the 1st order donor cell method (i.e. piecewise constant approximation).

3.1 Changes

3.1.1 Speed of Sound Calculation and Δt

For each cell we calculate the speed of sound for each material separately so that (2.52) becomes

$$ {}^{m}c_{i+1/2,k} = \sqrt{\frac{{}^{m}\gamma^{m}p_{i+1/2}}{{}^{m}\rho_{i+1/2}}} \tag{3.1} $$

where m is the material number. This is then used in (2.51) instead of $c_{i+1/2}$ to calculate Δt. Then (3.2) is used to calculate the average sound speed in

the cell.

$$\bar{c}_{i+1/2} = \sqrt{\frac{\sum_m {}^m F_{i+1/2}\, {}^m c^2_{i+1/2}\, {}^m \rho_{i+1/2}}{\sum_m {}^m F_{i+1/2}\, {}^m \rho_{i+1/2}}} \tag{3.2}$$

This sound speed is now used in the calculation of the artificial viscosity, q for the cell as a whole. ${}^m F_{i+1/2}$ is the fractional volume of material m in cell $i + 1/2$.

3.1.2 Lagrangian Phase

These equations are the 1-D simplification of the ones found in Debar [5] for the 2-D case.

Position, velocity and artificial viscosity are calculated for the node/cell as a whole. No changes are made to the equations of momentum (2.30), node motion (2.33) and artificial viscosity (2.38) except that where ρ and p are used in the single material code the average of these quantities is used here.

The internal energy, density and pressure are given by

$${}^m \varepsilon^{n+1}_{i+1/2} = {}^m \varepsilon^n_{i+1/2} - \frac{\Delta t \Delta u}{2\Delta x^n_{i+1/2}} \left(\frac{{}^m p^{n+\frac{1}{2}}_{i+1/2}}{{}^m \rho^n_{i+1/2}} + \frac{q^n_{i+1/2}}{\rho^n_{i+1/2}} \right) \tag{3.3}$$

$$\Delta u = u^n_{i+1} + u^{n+1}_{i+1} - u^n_i - u^{n+1}_i \tag{3.4}$$

$${}^m \rho^{n+1}_{i+1/2} = \frac{\Delta x^n_{i+1/2}\, {}^m \rho^n_{i+1/2}}{\Delta x^{n+1}_{i+1/2}} \tag{3.5}$$

$${}^m p^{n+1}_{i+1/2} = ({}^m \gamma - 1)\, {}^m \rho^{n+1}_{i+1/2}\, {}^m \varepsilon^{n+1}_{i+1/2} \tag{3.6}$$

3.1.3 Volume and Mass Fluxes

Here δV is the total amount of material to be advected which is in general made up of the constituent volumes of each material δV_A and δV_B,

$$\delta V = \delta V_A + \delta V_B$$

It is these volumes which need to be calculated so as to advect each material separately. When calculating the volume flux in a mixed cell two different cases can occur (see Figure 3.1). In the first case we have

$$\delta V > V_B$$

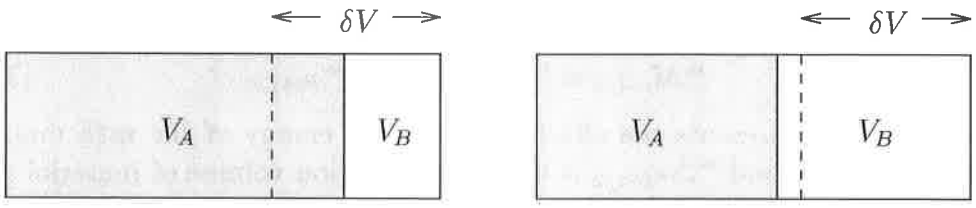

Figure 3.1: Calculation of volume flux in a mixed cell

so that

$$\delta V_B = V_B$$
$$\delta V_A = \delta V - V_B$$

and in the second case

$$\delta V < V_B$$

with

$$\delta V_B = \delta V$$
$$\delta V_A = 0$$

For generality in δV_A and δV_B we will use the notation ${}^m\delta V_i$ which represents the volume flux of material m through node i. The corresponding mass flux is analogous to (2.47), i.e.

$${}^m\delta M_i = {}^m\delta V_i {}^m f_i^\rho \tag{3.7}$$

3.1.4 Remap Step

The first thing to advect is the fractional volume

$${}^m\tilde{F}_{i+1/2} = \frac{\Delta x_{i+1/2}^{n+1} {}^m F_{i+1/2} + {}^m\delta V_i - {}^m\delta V_{i+1}}{\Delta x_{i+1/2}^n} \tag{3.8}$$

then density

$${}^m\tilde{\rho}_{i+1/2} = \frac{{}^m M_{i+1/2} + {}^m\delta V_i {}^m f_i - {}^m\delta V_{i+1} {}^m f_{i+1}}{{}^m\tilde{\Delta} x_{i+1/2}} \tag{3.9}$$

then energy

$${}^m\tilde{\varepsilon}_{i+1/2} = \frac{{}^m\varepsilon_{i+1/2} {}^m M_{i+1/2} + {}^m\delta M_i {}^m f_i - {}^m\delta M_{i+1} {}^m f_{i+1}}{{}^m\tilde{M}_{i+1/2}} \tag{3.10}$$

where

$$ {}^{m}M_{i+1/2} = {}^{m}F_{i+1/2}\Delta x_{i+1/2}{}^{m}\rho_{i+1/2} \tag{3.11} $$

Here ${}^{m}f_i$ represents the effective density or energy of the mth material through node i and ${}^{m}\tilde{\Delta}x_{i+1/2}$ is the post advection volume of material m in cell $i+1/2$ given by

$$ {}^{m}\tilde{\Delta}x_{i+1/2} = \Delta x_{i+1/2}{}^{m}\tilde{F}_{i+1/2} $$

Finally ψ_1 and ψ_2 can be advected as in the single material case since velocity is not calculated separately for each material.

The average densities and energies in each cell are given by

$$ \bar{\rho}_{i+1/2} = \sum_{k} {}^{m}F_{i+1/2}{}^{m}\rho_{i+1/2} \tag{3.12} $$

$$ \bar{\varepsilon}_{i+1/2} = \frac{\sum_{m} {}^{m}F_{i+1/2}{}^{m}\rho_{i+1/2}{}^{m}\varepsilon_{i+1/2}}{\sum_{m} {}^{m}F_{i+1/2}{}^{m}\rho_{i+1/2}} \tag{3.13} $$

3.2 Numerical Experiments

Figures 3.2, 3.3 and 3.4 show the numerical results for Sod's shock tube problem using the interface tracking techniques described above. This is exactly the same problem as in the single material case, i.e. the interface does not separate two different materials so $\gamma_L = \gamma_R$. All three cases, piecewise constant, linear and parabolic, show the same sharp profile across the contact discontinuity where it is only spread over a single cell. This cell just corresponds to the 'mixed' cell. In all three cases this method gives an overshoot in the internal energy profile near the contact discontinuity. As with the single material cases the piecewise linear and parabolic solutions are not noticeably different.

It was suggested by Jones [10] that if the interface was moved in the Sod problem so that it did not initially line up with the diaphragm and consequently the contact discontinuity, the results appear to show a small error in the piecewise linear case. Figure 3.5 shows a small glitch when the interface is in the rarefraction wave. He suggested that this may be due to the drop in accuracy to 1st order in a mixed cell and that not reducing to the donor cell method in a mixed cell and retaining accuracy may solve this problem.

To show this glitch more clearly Figure 3.6 shows the absolute error in the numerical solution for density compared to the exact solution for all three recovery steps. Four peaks occur in all three graphs and these coincide

with the head and tail of the rarefraction, the contact discontinuity and the shock wave. The height of the first two peaks and the spread of the errors in all the peaks can be reduced by either increasing the number of points or increasing the accuracy of the method. The height of the last two peaks, corresponding to the contact discontinuity and the shock wave, cannot be reduced and correspond to approximately half the height of the respective wave.

These errors are expected and will occur to a lesser or greater effect depending on the numerical scheme. The glitch occurs at about $x = 0.42$ and in all three cases is much smaller than the other numerical errors. Figure 3.7 focuses on this error alone by showing the absolute error of the numerical solution with the interface minus the numerical solution without the interface. This shows that both the piecewise linear and piecewise parabolic produce an error over twice as large as in the piecewise constant case. This gives some strength to the argument that maintaining the same accuracy of the scheme in a mixed cell improves this problem. However since no accuracy is lost in a mixed cell in the piecewise constant case this would probably not solve the problem completely. It is also clear that the error is not just confined to the mixed cell but spreads out to pollute the whole computation.

Until now all the numerical experiments on interface tracking have used only one material, i.e. $\gamma_L = \gamma_R$. Interface tracking is designed to deal with multimaterial flows and so to complete this section a Sod type problem is solved for all three recovery steps, see Figures 3.8, 3.9 and 3.10. As with all other experiments these show a similar improvement in resolution from piecewise constant to piecewise linear recovery and no noticeable improvement from piecewise linear to piecewise parabolic recovery.

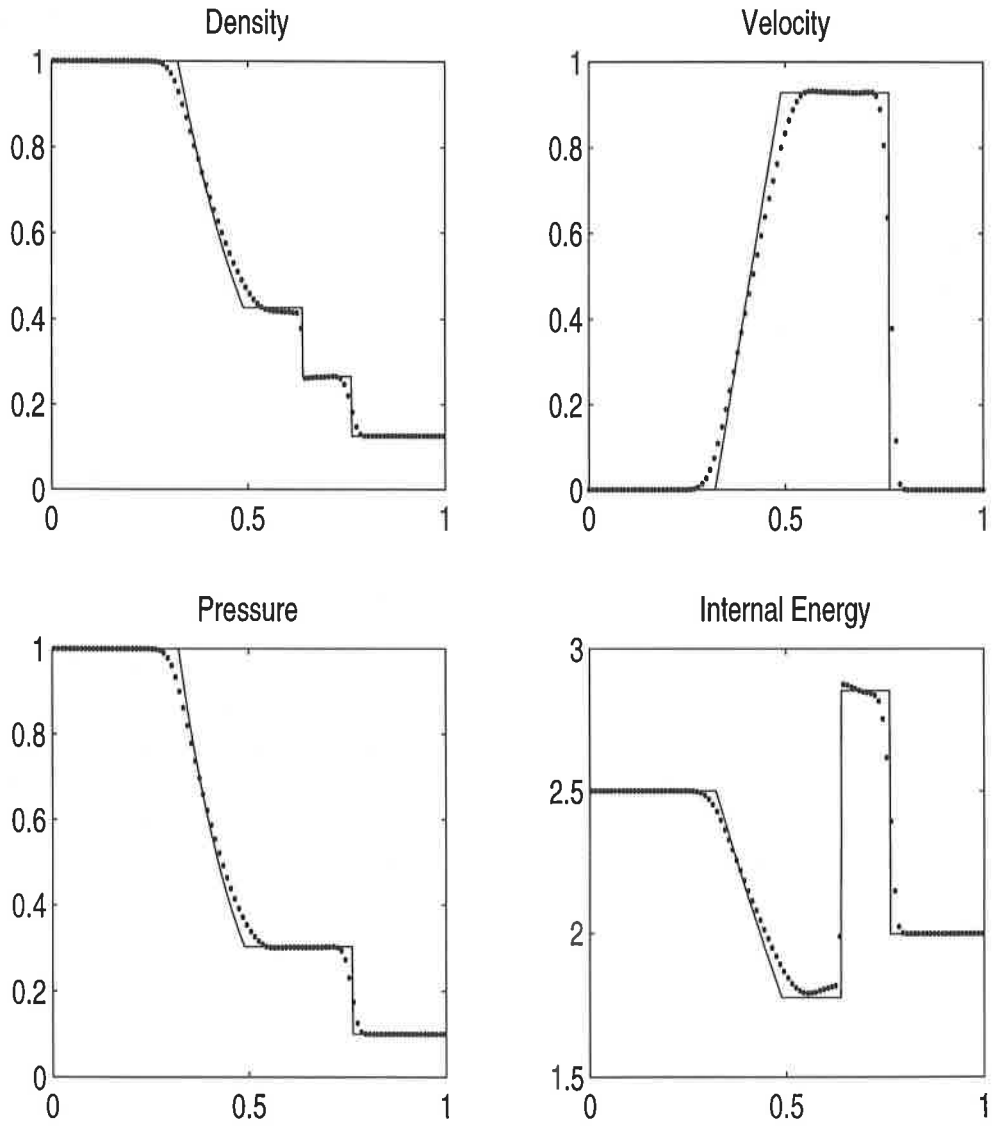

Figure 3.2: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise constant solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

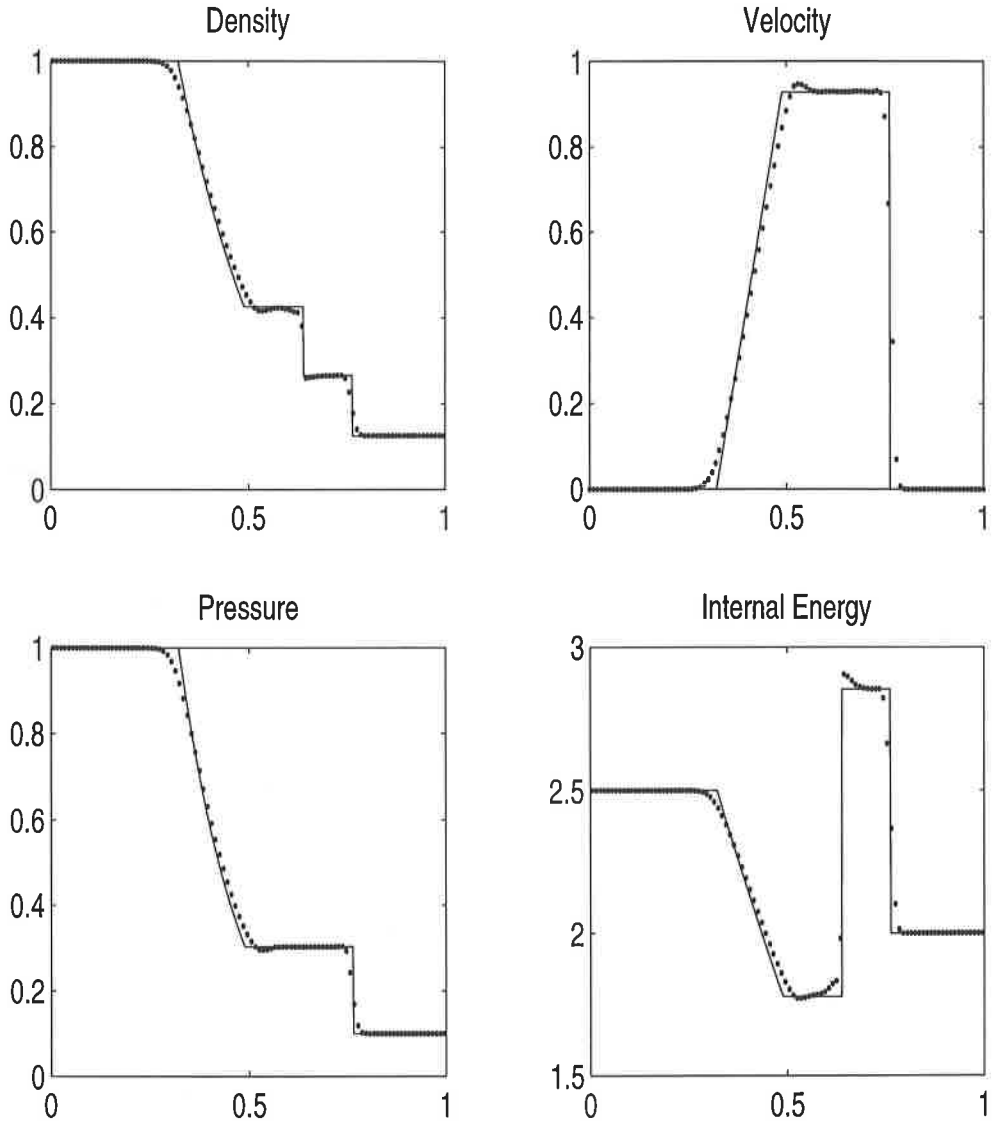

Figure 3.3: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise linear solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

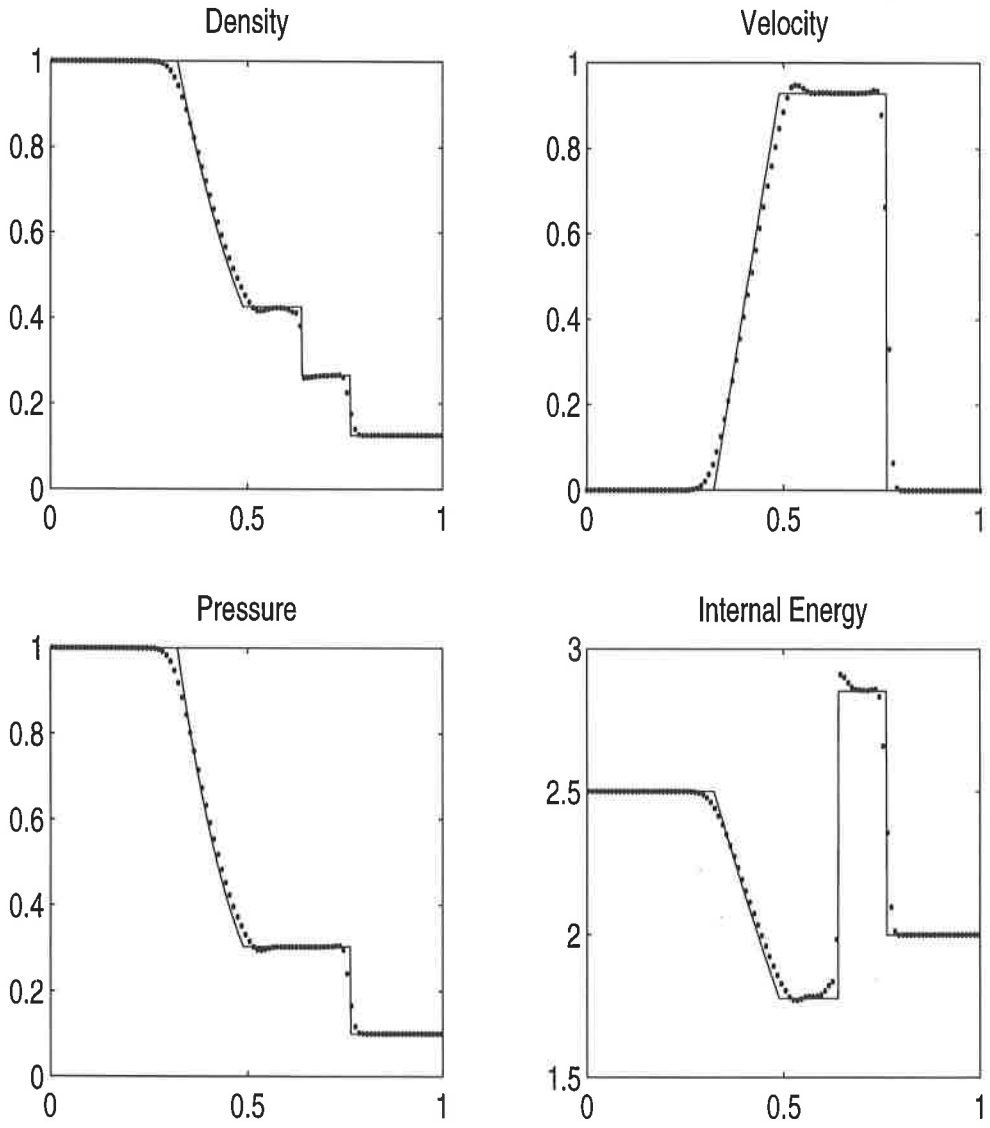

Figure 3.4: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise parabolic solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

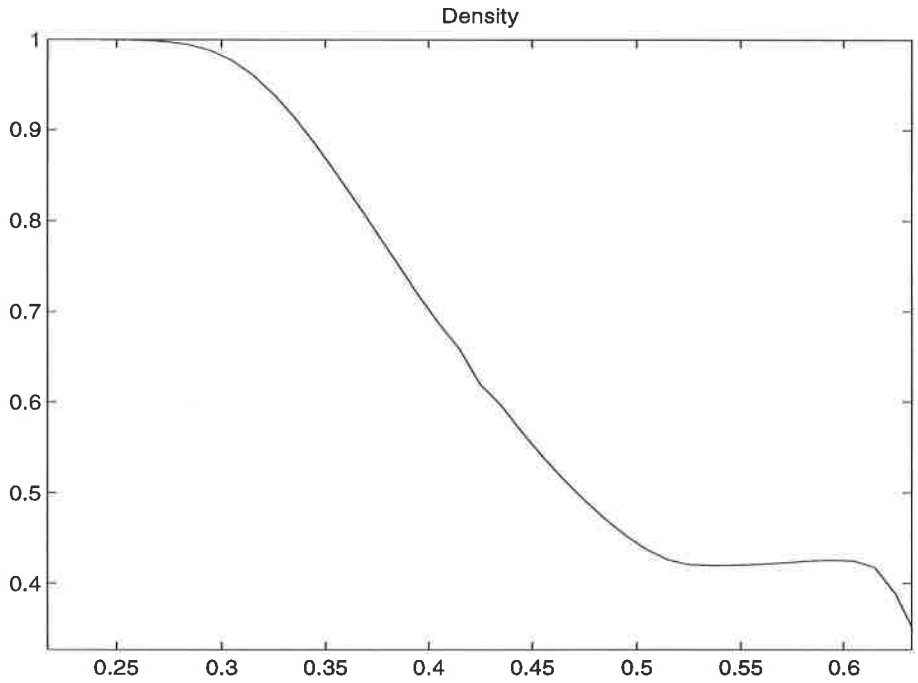

Figure 3.5: Error in rarefraction wave for Sod's shock tube problem with an interface using piecewise linear recovery. 100 computing cells. Interface is initially at 0.4 and diaphragm initially at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

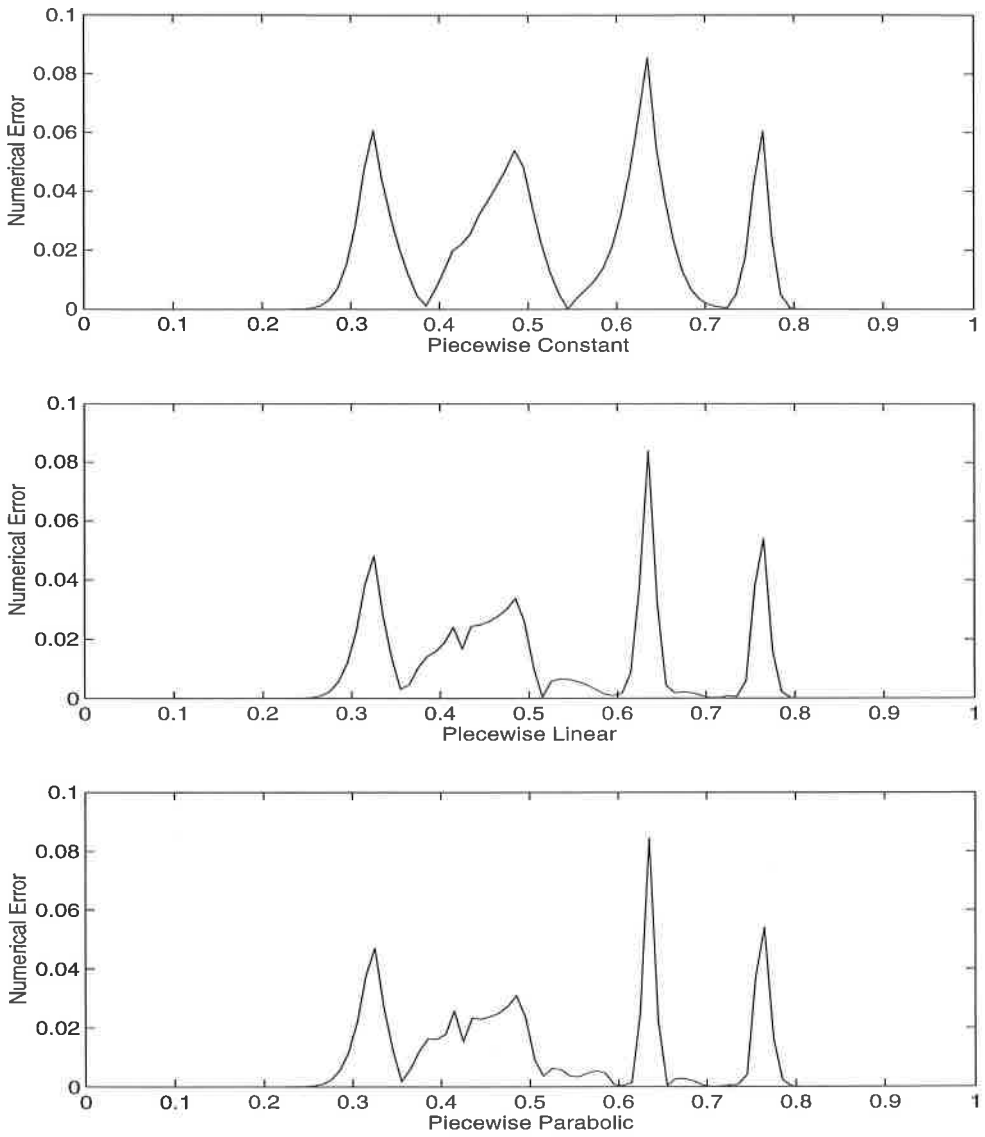

Figure 3.6: Numerical error ($|exact - numerical|$) for Sod's shock tube problem with an interface. 100 computing cells. Interface is initially at 0.4 and diaphragm initially at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

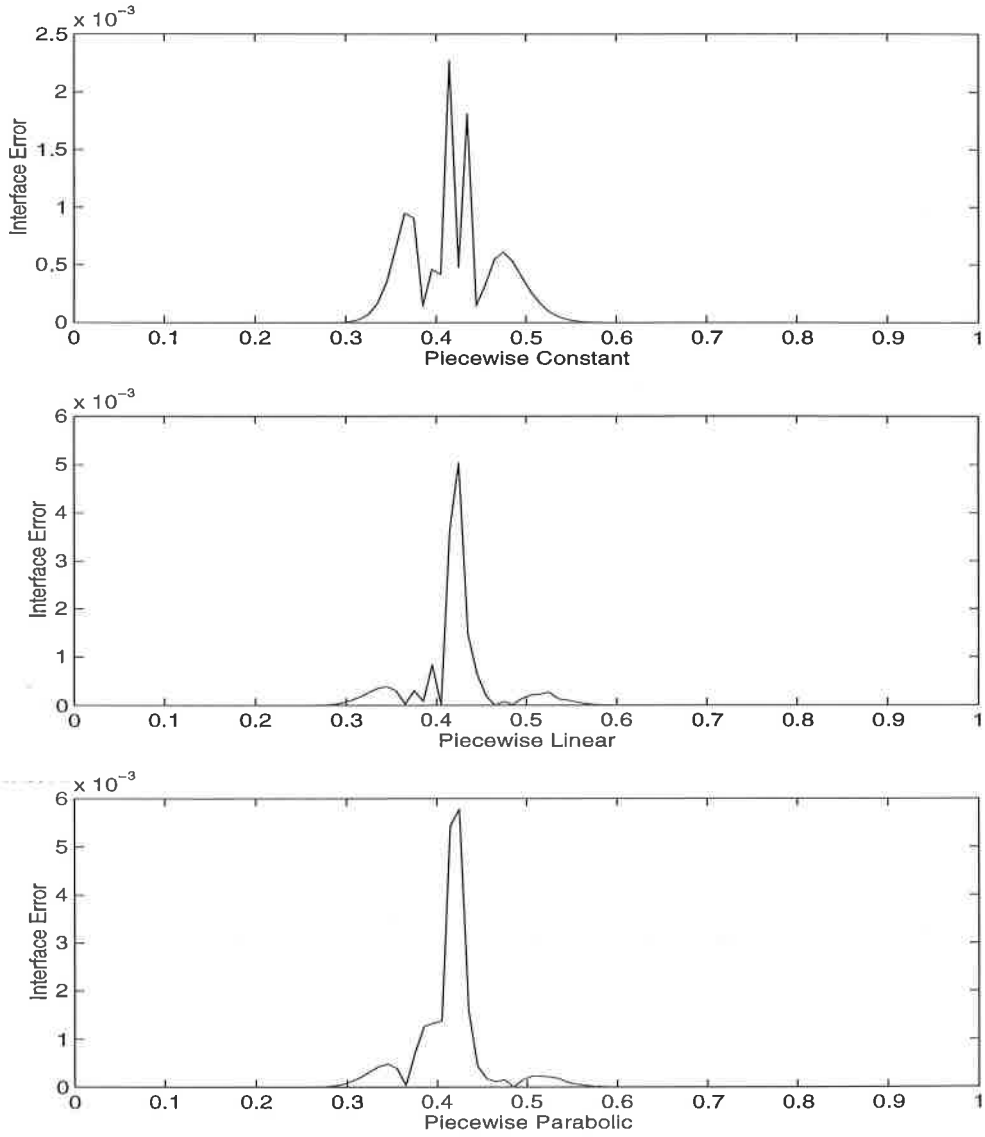

Figure 3.7: Interface error ($|with\ interface - without\ interface|$) for Sod's shock tube problem. 100 computing cells. Interface is initially at 0.4 and diaphragm initially at 0.5. Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.4$. Output time $t = 0.15$.

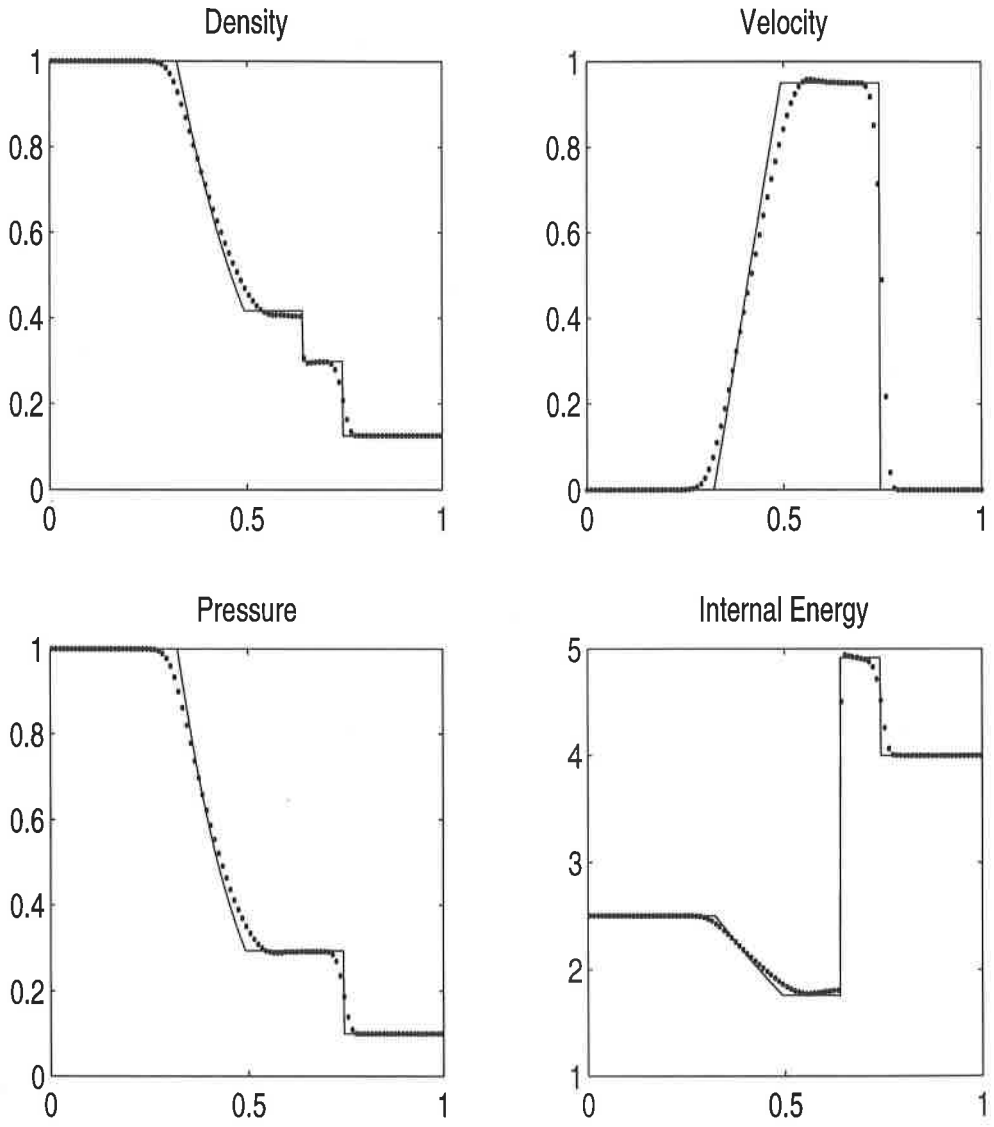

Figure 3.8: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise constant solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.2$. Output time $t = 0.15$.

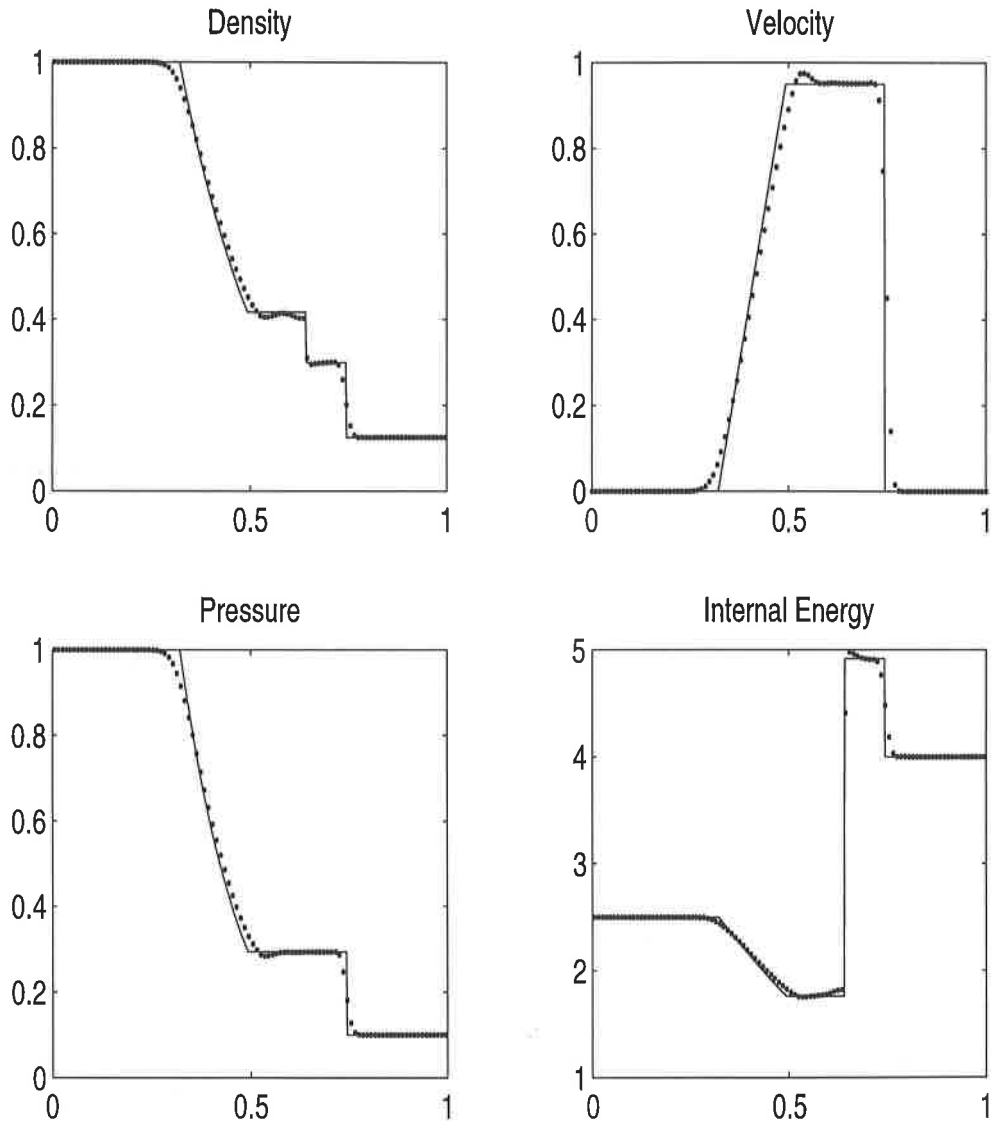

Figure 3.9: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise linear solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.2$. Output time $t = 0.15$.

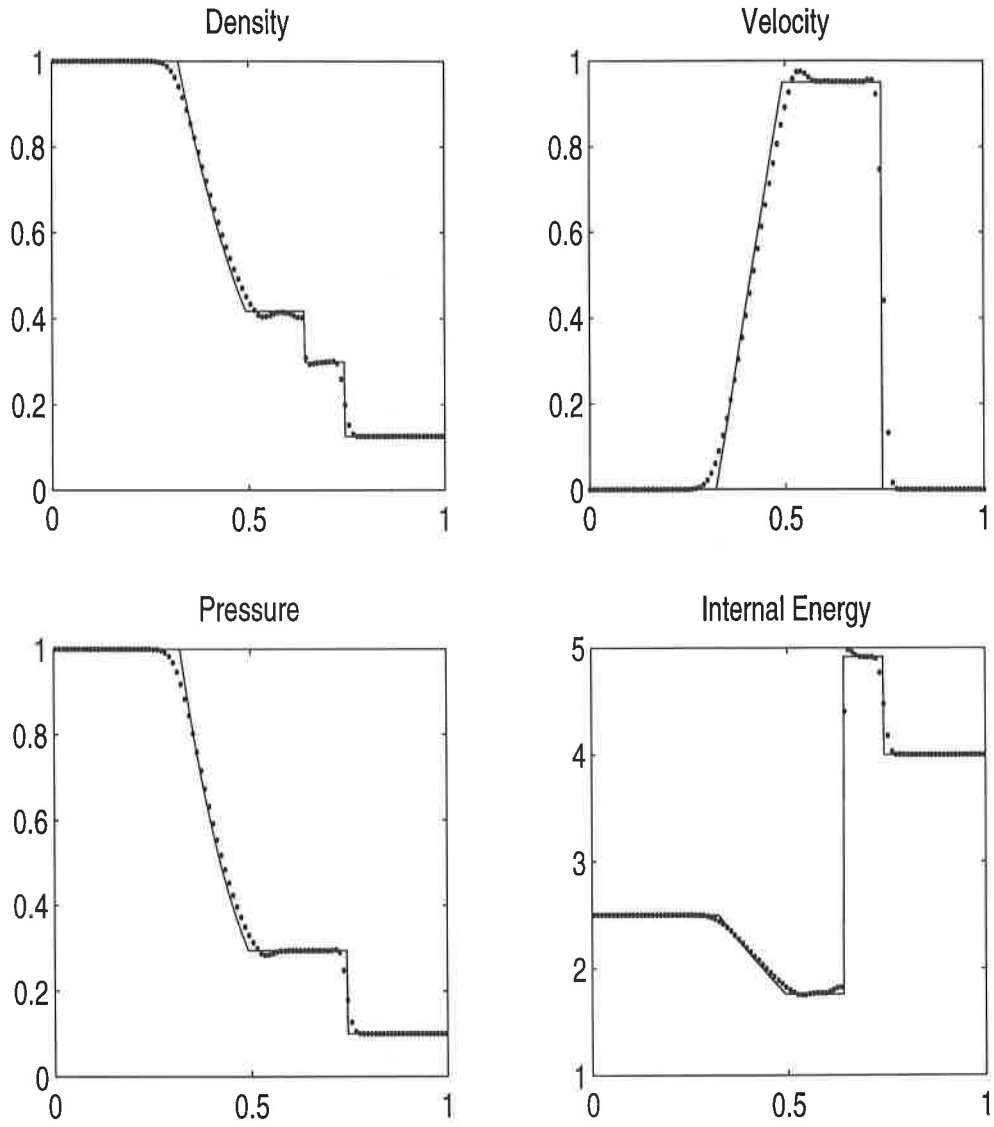

Figure 3.10: Sod's shock tube problem with an interface - Solid line is exact solution and dotted line is piecewise parabolic solution. 100 computing cells, interface and diaphragm initially coincide at 0.5, Hemp artificial viscosity with coefficients $c_0 = 1.5$ and $c_L = 0.06$. Initial left state: $\rho_L = 1.0$, $u_L = 0.0$, $p_L = 1.0$, $\gamma_L = 1.4$. Initial right state: $\rho_R = 0.125$, $u_R = 0.0$, $p_R = 0.1$, $\gamma_R = 1.2$. Output time $t = 0.15$.

Chapter 4

Conclusions

In Chapter 2 it was shown that in the linear advection case the 3rd order piecewise parabolic method performed considerably better than the 2nd order piecewise linear method which in turn is considerably better than the 1st order piecewise constant method. Results show that a square pulse advected through a mesh once with the piecewise linear method produces the same attenuation as advecting the same pulse through a mesh 10 times with the piecewise parabolic method. Unfortunately this large improvement does not follow when it is applied to solving the equations of gas dynamics. In this case the advection schemes are only a recovery step in the overall scheme. A 2nd order Lagrangian scheme is applied before all three recovery steps. It is probably this Lagrangian step that decreases the overall performance of the method with piecewise parabolic recovery. To test this would require the introduction of a 3rd order Lagrangian phase to see if this would improve the overall performance but this has not been dealt with here.

Chapter 3 uses interface tracking techniques in association with the previous methods to give a very sharp contact discontinuity profile, spread over only one cell even on a coarse grid. As in Chapter 2 all the results shown give little improvement in performance when piecewise parabolic recovery is used instead of piecewise linear. An error has been identified if the position of the interface is moved so as not to coincide with the diaphragm and consequently with the contact discontinuity. This error is not just confined to the mixed cell but also spreads out to pollute the flow. Although this error is small compared to other errors it could easily be magnified when the methods are extended to multidimensions. Unfortunately no cure has been found to this problem as yet and is an area of future research.

The other obvious area of future research is the extension to multidimensions of this work. This has been done on a quadrilateral mesh [5, 16, 17, 13] but little work has so far been done on triangular meshes.

Bibliography

[1] D.J. BENSON. Computational methods in Lagrangian and Eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering*, 99:235–394, 1992.

[2] D.J. BENSON. Momentum advection on a staggered mesh. *Journal of Computational Physics*, 100:143–162, 1992.

[3] P. COLELLA and P.R. WOODWARD. The piecewise parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics*, 54:174–201, 1984.

[4] R. COURANT and K.O. FRIEDRICHS. *Supersonic Flow and Shock Waves, Applied Mathematical Sciences.* Springer-Verlag, 1951.

[5] R.B. DEBAR. A method in 2-D Eulerian hydrodynamics. Technical Report UCID-19683, Lawrence Livermore Laboratory, 1974.

[6] F.N. FRITSCH and R.E. CARLSON. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17:238–246, 1980.

[7] S.K. GODUNOV. *Mat. Sb.*, 4:271, 1959.

[8] C.W. HIRT and B.D. NICHOLS. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201–225, 1981.

[9] J.M. HYMAN. Numerical methods for tracking interfaces. *Physica*, 12D:396–407, 1984.

[10] B. JONES. Private communication.

[11] B. VAN LEER. Towards the ultimate conservative difference scheme iv. A new approach to numerical convection. *Journal of Computational Physics*, 23:276–299, 1977.

[12] B. VAN LEER. Towards the ultimate conservative difference scheme v. A second-order sequel to Godunov's method. *Journal of Computational Physics*, 32:101–136, 1979.

[13] B.J. PARKER and D.L. YOUNGS. Two and three dimensional Eulerian simulation of fluid flow with material interfaces. Technical Report AWE Preprint 01/92, Atomic Weapons Establishment, 1992.

[14] S. STOKES. Eulerian methods with a Lagrangian phase in gas dynamics. Technical Report M.Sc. Thesis, University of Reading, 1995.

[15] P.R. WOODWARD and P. COLELLA. Numerical simulation of fluid flow, review article. *Journal of Computational Physics*, 54:115–173, 1984.

[16] D.L. YOUNGS. Time-dependent multi-material flow with large fluid distortion. *Numerical Methods for Fluid Dynamics, Academic Press*, pages 273–285, 1982.

[17] D.L. YOUNGS. An interface tracking methods for a 3D Eulerian hydrodynamics code. Technical Report AWRE/44/92/35, Atomic Weapons Research Establishment, 1987.