universität wien

DIPLOMARBEIT

Titel der Diplomarbeit

Simulation of Low Mach Number Fluids

Angestrebter akademischer Grad

Magister der Naturwissenschaften (Mag. rer. nat.)

Verfasser:	Natalie Happenhofer
Matrikel-Nummer:	0502382
Studienrichtung:	A 405 Mathematik
Betreuer:	Univ.-Prof. Dr. Herbert Muthsam

Wien, im Juni 2010

Contents

Chapter 1

Introduction

The foundational axioms of fluid dynamics are the conservation laws of mass, momentum and energy, leading to the well-known Navier-Stokes equations. Given this general ansatz, these equations can be applied to a broad range of problems, starting at modelling the flow through a tube in engineering sciences to describing stellar convection in astrophysics.

From a mathematical point of view, the Navier-Stokes equations constitute a nearly hyperbolic system of conservation laws, since we have very small viscosities. A variety of algorithms have been proposed to solve it numerically, each method designed to resolve certain physical features of the fluid.

In this diploma thesis, I use a fractional step method proposed by N. Kwatra et alii in [10]. The scheme divides the calculation in two parts: calculating the advective terms leading to intermediate values of the conserved quantities, and predicting the pressure at the next time step to correct those values.
This formulation is designed to effectively filter sound waves which transport a negligible amount of energy but move rather fast. To resolve them, little time steps have to be chosen, so alleviating the stringent CFL-condition imposed by the sound speed leads to significantly higher time steps.
We will especially focus on Low Mach Number Flows. The Mach Number Ma is defined as the ratio between the velocity of the fluid u and the sound speed c_s:

$$Ma = \frac{u}{c_s}$$

In general, fluids are considered to be 'Low Mach', if $Ma < 0.3$.
Traditional explicit schemes (ENO, WENO, etc.) require a large number of timesteps to simulate a period of interest in such scenarios. We will see that the gain in time by filtering the sound waves is quite enormous.

This fractional step method is implemented in the ANTARES code developed at the University of Vienna by H. Muthsam et alii, see [8]. ANTARES is a highly parallelized hydrodynamic code used to perform simulations of solar granulation, cepheids, A-stars and the stellar interior, especially convection, diffusive convection and magnetohydrodynamics.
Simulations are run on high performance clusters like the LRZ in Munich, and, more recently, at the Vienna Scientific Cluster.

Chapter 2

The Equations of Hydrodynamics

In this chapter we develop the basic equations of fluid mechanics. These equations are derived from the conservation laws of mass, momentum and energy. For all calculations standard Euclidian coordinates are used and the physical quanitites are given in CGS units.

Let D be an arbirary volume D in the (two- or three dimensional) space in which our fluid flow takes place.
Let $\mathtt{x} \in D$ be a point in D, $\mathtt{x} = (x, y, z)$, and consider a particle of fluid moving through x at time t. Imagine a particle of dust suspendend in the fluid; this particle traverses a well-defined trajectory. Be $\vec{u}(\mathtt{x}, t)$ the velocity of the particle, we find that $\vec{u}$ is a vector field on D. We call $\vec{u} = (u(\mathtt{x}, t), v(\mathtt{x}, t), w(\mathtt{x}, t))$ the **velocity field** of the fluid, $[\vec{u}] = cm/s$.

Consider a subregion W of D subject to the fluid flow, i.e: As the particles at the initial position $\mathtt{x}$ travel, W will change its shape due to the trajectories of its particles.
For better legibility, we will carry out the following considerations writing W instead of $W(t)$ and denoting by $\mathtt{x}$ the coordinates of the points in W, bearing in mind that the coordinates in $W(t)$ are actually dependent on their initial positions. See Appendix A.3 for more details.

2.1 Conservation of Mass

Let $\rho(\mathtt{x}, t)$ denote the mass density of the fluid,$[\rho] = g$. To perform analytic calculations, we assume that $\rho(\mathtt{x}, t)$ and $\vec{u}$ are continuous.
The mass of fluid in W at time t is given by

$$m(W, t) = \int_W \rho(\mathtt{x}, t) dV \tag{2.1.1}$$

The principle of conservation of mass says that the mass in the volume does not change with time:

$$\frac{d}{dt}m(W,t) = 0 \tag{2.1.2}$$

Inserting equation (2.1.1), this reads

$$\frac{d}{dt}\int_W \rho dV = 0 \tag{2.1.3}$$

Using the Theorem 1.3.3 of Appendix A, we can rewrite this as:

$$\int_W \left[\frac{\partial \rho}{\partial t} + div(\rho\vec{u})\right] dV = 0 \tag{2.1.4}$$

Since W was arbitrarily chosen, we arrive at the following PDE:

$$\frac{\partial \rho}{\partial t} + div(\rho\vec{u}) = 0 \tag{2.1.5}$$

This is the differential form of the law of conservation of mass, also called the **continuity equation**. It holds in this form if ρ and $\vec{u}$ are $\mathcal{C}^1$.

2.2 Balance of momentum

Definition 2.2.1. Be $g(\mathtt{x}(t),t)$ an arbitrary function of position and time. Taking the derivative along a path moving with velocity $\vec{u}$, the chain rule yields

$$\frac{d}{dt}g(\mathtt{x}(t),t) = \partial_t g + \vec{u}\cdot\nabla g =: \frac{Dg}{Dt}(\mathtt{x}(t),t) \tag{2.2.1}$$

$\frac{Dg}{Dt}$ is called the **material derivative** of g.

The momentum density of the fluid is $\rho\vec{u}$. Therefore, the total momentum in W is the volume integral over $\rho\vec{u}$:

$$Mom(W,t) = \int_W \rho\vec{u}dV \tag{2.2.2}$$

Let $f(\mathtt{x},t)$ denote the force exerted on location $\mathtt{x} \in W$ at time t. The total force exerted on W is given by

$$F(W,t) = \int_W f(\mathtt{x},t)dV \tag{2.2.3}$$

Newton's Second Law of Motion states that the rate of change of momentum is equal

to the net force exerted on the object considered:

$$\frac{d}{dt}Mom(W,t) = F(W,t) \tag{2.2.4}$$

Inserting expressions (2.2.2) - (2.2.3), this becomes:

$$\frac{d}{dt}\int_W \rho\vec{u}dV = \int_W f(\mathtt{x},t)dV \tag{2.2.5}$$

Using the Transport Theorem 1.3.2, we can rewrite this to

$$\int_W \rho\frac{D\vec{u}}{Dt}dV = \int_W f(\mathtt{x},t)dV \tag{2.2.6}$$

Basically, we consider two different types of stresses exerted on our fluid in W: **body forces** B_W and **surface forces** $S_{\partial W}$.

$$\int_W \rho(\mathtt{x},t)\frac{D\vec{u}}{Dt}dV = B_W + S_{\partial W} \tag{2.2.7}$$

We will examine them now more closely.

2.2.1 Body forces

A body force is a force that acts throughout the volume of a body. Gravity and electromagnetic forces are examples of such.
If $\vec{b}(\mathtt{x},t)$ denotes the given body force per unit mass, the total body force $\vec{B}_W$ is

$$\vec{B}_W = \int_W \rho\vec{b}dV \tag{2.2.8}$$

2.2.2 Surface forces

A surface force is a force that acts across a surface element of the region W.
Let $\vec{t}(\mathtt{x},t,\vec{n})$ denote the surface force on $\mathtt{x} \in \partial W$ at time t, $\vec{n}$ be the outward unit normal. The total surface force is given by

$$S_{\partial W} = \int_{\partial W} t(\mathtt{x},t,\vec{n})dS \tag{2.2.9}$$

Theorem 2.2.2. *Choose a region W_L such that $Vol(W_L) = L^3$ Then the shear forces locally annihilate each other:*

$$\lim_{L\to 0}\int_{\partial W_L} \|t(\mathtt{x},t,\vec{n})\|dS = 0 \tag{2.2.10}$$

Proof. Consider the balance of momentum (2.2.7):

$$\int_{\partial W_L} t(\mathtt{x}, t, \vec{n}) dS = \int_W \rho(\mathtt{x}, t) \frac{D\vec{u}}{Dt} - \rho\vec{b} dV \tag{2.2.11}$$

We choose a suitable norm and apply the triangle inequality onto the right hand side:

$$\| \int_{\partial W_L} t(\mathtt{x}, t, \vec{n}) dS \| = \| \int_W \rho(\mathtt{x}, t) \frac{D\vec{u}}{Dt} - \rho\vec{b} dV \| \leq \int_W \| \rho(\mathtt{x}, t) \frac{D\vec{u}}{Dt} - \rho\vec{b} \| dV \tag{2.2.12}$$

Assuming that the expression $\rho(\mathtt{x}, t)\frac{D\vec{u}}{Dt} - \rho\vec{b}$ is bounded we get

$$\int_W \| \rho(\mathtt{x}, t) \frac{D\vec{u}}{Dt} - \rho\vec{b} \| dV \leq CL^3 \tag{2.2.13}$$

where C is constant.
Combining the above equations and dividing by L^2 leads to

$$\| \int_{\partial W_L} t(\mathtt{x}, t, \vec{n}) dS \| \to 0 \text{ if } L \to 0 \tag{2.2.14}$$

Thus we arrive at

$$\int_{\partial W_L} \| t(\mathtt{x}, t, \vec{n}) \| dS \to 0 \text{ if } L \to 0 \tag{2.2.15}$$

□

Theorem 2.2.3. *There is a matrix-valued function $\mathcal{T} : W \times (0, \infty) \to M_3(\mathbb{R})$ which satisfies*

$$t(\mathtt{x}, t, \vec{n}) = \mathcal{T}(\mathtt{x}, t) \cdot \vec{n} \tag{2.2.16}$$

Proof. Be $R_L(\mathtt{x}, \vec{n})$ a tetraeder with annotations shown in Figure (2.1).
The surfaces S_i $(i = 1, 2, 3)$ are orthogonal to the canonical unit vectors e_i $(i = 1, 2, 3)$ and the distance of $\mathtt{x}$ is such that the area of S equals L^2.
Let us have a look at the force applied to the lateral surfaces with area S_i $(i = 1, 2, 3)$. Using the mean value theorem for integration we get

$$\int_{S_i} \vec{t}(\mathtt{x}, t, \vec{n}) dS = S_i \vec{t}(\mathtt{x}, t, -\vec{e_i}), \mathtt{x}_i \in S_i, i = 1, 2, 3$$

$$\int_S \vec{t}(\mathtt{x}, t, \vec{n}) dS = S\vec{t}(\mathtt{x}_0, t, \vec{n}), \mathtt{x}_0 \in S$$

Since the S_i are the projections of S onto the i^{th} coordinates, we know that

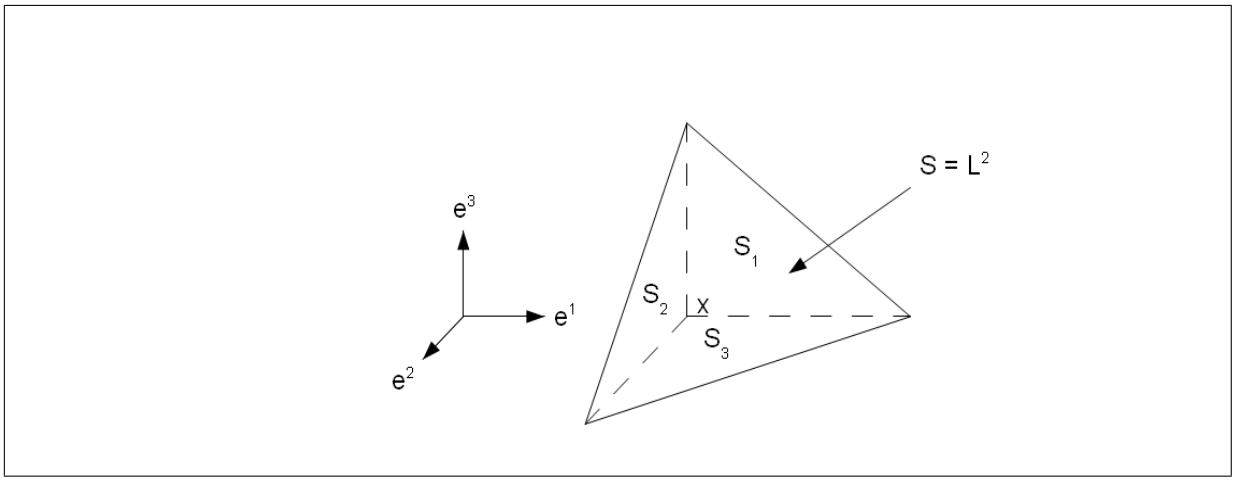


Figure 2.1: Tetraeder

$$S_i = (\vec{n} \cdot \vec{e}_i)S = n_i S$$

Theorem 2.2.2 states that locally the shear forces annihilate each other:

$$\sum_{i=1}^{3} n_i \vec{t}(\mathtt{x}, t, -\vec{e}_i) + \vec{t}(\mathtt{x}_0, t, \vec{n}) = 0 \tag{2.2.17}$$

Newton's Third Law of Motion says that to every action there is an equal opposite reaction:

$$\vec{t}(\mathtt{x}, t, -\vec{n}) = -\vec{t}(\mathtt{x}, t, \vec{n}) \tag{2.2.18}$$

So equation (2.2.17) becomes

$$\vec{t}(\mathtt{x}_0, t, \vec{n}) = \sum_{i=1}^{3} n_i \vec{t}(\mathtt{x}, t, \vec{e}_i) \tag{2.2.19}$$

We define $\mathcal{T} \in \mathbb{R}_3$ by

$$\mathcal{T}_{ij} = t_j(\mathtt{x}, t, \vec{e}_i)$$

Thus we can rewrite equation (2.2.19) as

$$\vec{t}(\mathtt{x}, t, \vec{n}) = \vec{n} \cdot \mathcal{T}(\mathtt{x}, t)$$

□

Definition 2.2.4. We refer to $\mathcal{T}(\mathtt{x}, t)$ the **stress tensor**, whereas $t(\mathtt{x}, t, \vec{n})$ is called the **stress vector**.

Remark 2.2.5. Another physical principle is the conservation of angular momentum in a closed system. This leads to the deduction, that the stress tensor $\mathcal{T}(\mathtt{x}, t)$ is symmetric.

For a detailed derivation, see [13, chap. 2].

Surface forces can be decomposed into pressure forces acting orthogonally on the surface and shear forces (viscous forces):

$$\mathcal{T}(\mathtt{x}, t) = PI + \sigma \tag{2.2.20}$$

I denotes the identity matrix.

Definition 2.2.6. The normal component P is called the **pressure**, whereas σ is referred to as **viscous stress tensor** or **viscosity tensor**.

Consider a fluid where no tangential forces are at work. Consequentially,the surface forces are described by

$$t(\mathtt{x}, t, \vec{n}) = p(\mathtt{x}, t, \vec{n}) \cdot \vec{n} \tag{2.2.21}$$

Theorem 2.2.7. *P does not depend on the unit normal $\vec{n}$.*

Proof. In view of Theorem 2.2.3 we get

$$\vec{n} \cdot \mathcal{T}(\mathtt{x}, t) = -P(\mathtt{x}, t, \vec{n})\vec{n}$$

Choosing $\vec{n} = \vec{e}_i$ and taking the inner product with e_j this becomes

$$\mathcal{T}_{ij} = -P(\mathtt{x}, t, \vec{n})$$

We now use the definition of $T(\mathtt{x}, t)$:

$$-n_j P(\mathtt{x}, t, \vec{e}_i) = -n_j P(\mathtt{x}, t, \vec{n})$$

Division by $(-n_j)$ yields our statement.

□

The viscous stress tensor

We will now examine σ more closely. The following derivation is taken from [13, chap. 3].

Definition 2.2.8. The **deformation tensor** is defined by

$$D := \frac{1}{2}(\nabla\vec{u} + \nabla\vec{u}^t) \tag{2.2.22}$$

Assumption 1: Inner friction forces occur if parts of the fluid move with different velocities. So σ has to depend on the spatial derivatives of the velocity $\vec{u}$. If the velocity gradients

are small, it is justified to take just the the first spatial derivatives of $\vec{u}$ into account (see [11], p. 53).

$$\sigma_{ij} = \sum_{k,l} C_{ijkl} \frac{\partial u_k}{\partial u_l} \qquad i,j = 1,2,3 \tag{2.2.23}$$

C is a tensor of fourth order.
Since σ is symmetric (see remark 2.2.3),

$$C_{ijkl} = C_{jikl} \qquad i,j = 1,2,3 \tag{2.2.24}$$

holds.
Assumption 2: We assume that the material is homogenous. Thus the coefficients C_{ijkl} are constant.

$$C_{ijkl} \equiv const. \tag{2.2.25}$$

If the fluid rotates like a rigid body, the velocity is location-dependent, but we will not observe any friction forces. We require the viscosity tensor to vanish in this case.
Revolution about the z-axis is mathematically described by

$$\vec{u} = \omega(-y, x, 0)$$

This leads to

$$C_{ij21} = C_{ij12} \qquad i,j = 1,2,3 \tag{2.2.26}$$

Analogous considerations of revolution about the x- and y-axis lead to

$$C_{ij32} = C_{ij23} \qquad i,j = 1,2,3 \tag{2.2.27}$$

$$C_{ij31} = C_{ij13} \qquad i,j = 1,2,3 \tag{2.2.28}$$

So far, our ansatz (2.2.23) reads

$$\sigma_{ij} = C_{ij11}\frac{\partial u}{\partial x} + C_{ij22}\frac{\partial v}{\partial y} + C_{ij33}\frac{\partial w}{\partial z} + C_{ij12}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) + C_{ij13}(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}) + C_{ij23}(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}) \tag{2.2.29}$$

We note that σ just depends on the deformation tensor D.
Assumption 3: The fluid is isotropic, i.e. it is uniform in all directions.
In terms of mathematics this is expressed by invariance orthogonal transformations $S \in O_3$, $O_3 = \{S \in M_3(\mathbb{R}) \mid det(S) = \pm 1\}$

Remark 2.2.9. $S \in O_3$ satisfies the following property: $S^t = S^{-1}$

Let us apply such an orthogonal transformation S. We need to transform the spatial coordinates $\mathtt{x}$

$$\mathtt{x}' = S\mathtt{x}$$

the velocity field $\vec{u}$

$$\vec{u}' = S\vec{u} \tag{2.2.30}$$

and the frictional force denoted by $\vec{f} = \vec{n} \cdot \sigma$. The latter is transformed onto a surface element with unit normal $\vec{n}$:

$$\vec{f}' = S\vec{f} \tag{2.2.31}$$

$$\vec{f}' = \vec{n}'\sigma' \tag{2.2.32}$$

$$\vec{n}' = S\vec{n} \tag{2.2.33}$$

Elaborating these equations a bit further leads to:

$$f_i' = \sum_{j=1}^{3} S_{ij}(\vec{n} \cdot \sigma)_j = \sum_{j=1}^{3}\sum_{k=1}^{3} S_{ij} n_k \sigma_{kj} = \sum_{j=1}^{3}\sum_{k=1}^{3}\sum_{l=1}^{3} S_{ij} S_{lk} n_l' \sigma_{kj} = \tag{2.2.34}$$

$$\sum_{l=1}^{3} n_l' \sum_{j=1}^{3}\sum_{k=1}^{3} S_{lk}\sigma_{kj}S_{ji}^t = (\vec{n'} \cdot (S\sigma S^t))_i, i = 1,2,3 \tag{2.2.35}$$

Since $\vec{n}$ - and therefore $\vec{n}'$ - is arbitrarily chosen we deduce

$$\sigma' = S\sigma S^t \tag{2.2.36}$$

Componentwise, this reads

$$\sigma_{ij}' = \sum_{k=1}^{3}\sum_{l=1}^{3} S_{ik}\sigma_{kl}S_{jl} = \sum_{l=1}^{3}\sum_{k=1}^{3}\sum_{m=1}^{3}\sum_{n=1}^{3} S_{ik}S_{jl}C_{klmn}\frac{\partial u_m}{\partial x_n}, i,j = 1,2,3 \tag{2.2.37}$$

Applying S on the Jacobian of $\vec{u}$ we get

$$\nabla'\vec{u}' = S\nabla\vec{u}S^t \tag{2.2.38}$$

Since we assume that the dependance of σ of $\nabla\vec{u}$ is invariant under the rotation performed,

we now calculate σ using (2.2.23) and (2.2.38)

$$\sigma'_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} C_{ijkl}\frac{\partial u'_k}{\partial x'_l} = \sum_{l=1}^{3}\sum_{k=1}^{3}\sum_{m=1}^{3}\sum_{n}^{3} S_{km}S_{ln}C_{ijkl}\frac{\partial u_m}{\partial x_n}, \quad i,j = 1,2,3 \tag{2.2.39}$$

A comparison of (2.2.37) and (2.2.39) leads to

$$\sum_{k=1}^{3}\sum_{l=1}^{3} S_{ij}S_{jl}C_{klmn} = \sum_{k=1}^{3}\sum_{l=1}^{3} S_{km}S_{ln}C_{ijkl}, \quad i,j,m,n = 1,2,3 \tag{2.2.40}$$

This is to hold for every $S \in O_3$. To deduce equations for the components C_{ijkl}, we choose explicit orthogonal matrices:

$$S_1 = \begin{pmatrix} (-1)^{k_1} & 0 & 0 \\ 0 & (-1)^{k_2} & 0 \\ 0 & 0 & (-1)^{k_3} \end{pmatrix}$$

Inserting S_1 into (2.2.40) and choosing suitable values for k_1, k_2, k_3 leads to

$$\begin{aligned} C_{1212} &= C_{1221} = C_{2121} = C_{2112} \\ C_{1313} &= C_{1331} = C_{3131} = C_{3113} \\ C_{2323} &= C_{2332} = C_{3232} = C_{3223} \end{aligned}$$

Next we choose

$$S_2 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

and some special quadruples (i,j,m,n) in (2.2.40). The quadruples and the resulting equations are listed below.

$$\begin{aligned} (2,2,1,1) \quad & C_{1111} = C_{2222} \\ (1,1,3,3) \quad & C_{1111} = C_{3333} \\ (1,1,1,1) \quad & C_{1122} = C_{3311} \\ (2,2,2,2) \quad & C_{1122} = C_{2233} \\ (1,1,2,2) \quad & C_{1133} = C_{3322} \\ (2,2,3,3) \quad & C_{1133} = C_{2211} \\ (1,2,1,3) \quad & C_{1221} = C_{3113} \\ (1,3,2,3) \quad & C_{1331} = C_{3223} \end{aligned}$$

Last but not least we take

$$S_3 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

and $(i, j, k, m) = (1, 1, 1, 1)$. The resulting equation is

$$C_{1122} = C_{2211}$$

So C has at most three non-vanishing coefficients:

$$\begin{aligned} \kappa &= C_{iiii} && i = 1, 2, 3 \\ \lambda &= C_{iijj} && i \neq j \\ \mu &= C_{ijij} = C_{ijji} && i \neq j \end{aligned}$$

Therefore the viscous stress tensor is given by

$$\sigma = \lambda(\nabla \cdot \vec{u})I_3 + 2\mu D + (\kappa - \lambda - 2\mu)diag(\nabla \vec{u}) \tag{2.2.41}$$

Naturally, we require equation (2.2.37) to hold. In general we find that

$$I_3' = SI_3S^t = I_3$$

and

$$D' = \frac{1}{2}(\nabla' \vec{u}' + (\nabla' \vec{u}')^t) = \frac{1}{2}(S\nabla \vec{u}S^t + (S\nabla \vec{u}S^t)^t) = \frac{1}{2}S(\nabla \vec{u} + (\nabla \vec{u})^t)S^t = SDS^t$$

but

$$diag(\nabla' \vec{u'}) = diag(S\nabla \vec{u}S^t) \neq Sdiag(\nabla \vec{u})S^t$$

Thus

$$\kappa - \lambda - 2\mu = 0$$

and the viscous stress tensor reduces to

$$\sigma = \lambda(\nabla \vec{u})I + 2\mu D \tag{2.2.42}$$

Expressing σ in terms of μ and $\eta := \lambda + \frac{2}{3}\mu$ leads to

$$\sigma = 2\mu(D - \frac{1}{3}(\nabla \cdot \vec{u})I_3) + \eta(\nabla \cdot \vec{u})I_3 \tag{2.2.43}$$

μ is the coefficient of **shear viscosity** and η is the coefficient of **bulk viscosity**. These are also called *the* coefficient of viscosity and the *second* coefficient of viscosity respectively. In general, μ and η depend on the pressure P and the temperature T.

We remember now our equation for the balance of momentum:

$$\int_W \rho(\mathtt{x},t)\frac{D\vec{u}}{Dt}dV = \int_W \rho\vec{b}dV + \int_{\partial W} \mathcal{T}(\mathtt{x},t)dS \tag{2.2.44}$$

Due to our considerations above and Gauss's Divergence Theorem 1.2.1, we write this as as

$$\int_W \rho(\mathtt{x},t)\frac{D\vec{u}}{Dt}dV = \int_W \rho\vec{b}dV + \int_{\partial W} P\vec{n} + \sigma\vec{n}dS = \tag{2.2.45}$$

$$\int_W \rho\vec{b}dV + \int_W \nabla P dV + \int_W \nabla\sigma dV \tag{2.2.46}$$

Since W was arbitrarily chosen, we can now formulate the differential form of **balance of momentum**:

$$\rho\frac{D\vec{u}}{Dt} = \nabla P + \nabla\sigma + \rho\vec{b} \tag{2.2.47}$$

2.3 Conservation of Energy

Energy exists in many forms. Here we concentrate on the two most basic ones: the specific thermal (internal) energy $e(\mathtt{x},t)$ and the specific kinetic energy $e_{kin}(\mathtt{x},t) = \frac{1}{2}\vec{u}^2(\mathtt{x},t)$. The sum of these is the specific total energy: $e_{tot}(\mathtt{x},t) = e(\mathtt{x},t) + e_{kin}(\mathtt{x},t)$, $[e] = [e_{kin}] = [e_{tot}]$ $= erg/K$. The total energy contained in W is the volume integral of ρe_{tot}:

$$E(W,t) = \int_W \rho e_{tot}dV = \int_W \rho(e + \frac{1}{2}\vec{u}^2)dV \tag{2.3.1}$$

To formulate the conservation of energy, we need to take into account the work that the exterior does on W and a certain thermal conduction $h(x,t)$.

$$\frac{\partial}{\partial t}\int_W \rho e_{tot}dV = \int_W \underbrace{\rho\vec{f}\cdot\vec{u}}_{\text{work done by body forces}} dV + \int_{\partial W} \underbrace{-h\cdot\vec{u}}_{\text{heat conduction}} dS + \int_{\partial W} \underbrace{\vec{t}\cdot\vec{u}}_{\text{work done by surface forces}} dS \tag{2.3.2}$$

Theorem 2.3.1. *There is a vector* $\vec{q}(\mathtt{x}, t)$ *such that*

$$h = \vec{n} \cdot \vec{q}(\mathtt{x}, t)$$

holds.

Proof. The proof is analogous to the proof of theorem 2.2.3. See [13]. □

$q(\mathtt{x}, t)$ is the **thermal flux density**, $[\vec{q}] = J/s^{-3}$. **Newton's law of cooling** states that the rate of heat loss of a body is proportional to the difference in temperatures between the body and its surroundings:

$$q = \kappa \nabla T$$

The **thermal conductivity** κ is a material-dependent variable, $[\kappa] = J/s^2K$, whereas T denotes the temperature, $[T] = K$. See [11], p. 211, for more details.

We rewrite equation (2.3.2) using Theorem 2.3.1:

$$\frac{\partial}{\partial t} \int_W \rho e_{tot} dV = \int_W \rho \vec{b} \cdot \vec{u} dV + \int_{\partial W} (-q + T \cdot \vec{u}) \cdot \vec{n} dS \tag{2.3.3}$$

Applying Gauss's Divergence Theorem 1.2.1 and the Transport Theorem 1.3.2 this leads to

$$\int_W \rho \frac{De_{tot}}{Dt} = \int_W \rho \vec{b} \cdot \vec{u} + \int_W \nabla(-q + T \cdot \vec{u}) \tag{2.3.4}$$

Remembering that W was arbitrarily chosen, we deduce

$$\rho \frac{De_{tot}}{Dt} = \rho \vec{b} \cdot \vec{u} dV + \nabla(-q + T \cdot \vec{u}) dS \tag{2.3.5}$$

This is the **differential form of the conservation of energy**.

By simple calculus we can rewrite (2.3.5) to

$$\frac{\partial(\rho e_{tot})}{\partial t} = -\nabla(u(\rho e_{tot} + P)) + \nabla \cdot (\vec{u} \cdot \sigma) + \rho \vec{b} - \nabla q \tag{2.3.6}$$

2.4 The Euler and Navier-Stokes Equations

Summing up the derived equations, we get the **Navier Stokes** equations for compressible fluids:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{2.4.1}$$

$$\frac{\partial (\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u} + P) - \nabla \cdot \sigma = \rho \vec{b} \tag{2.4.2}$$

$$\frac{\partial (\rho e_{tot})}{\partial t} + \nabla \cdot (u(\rho e_{tot} + P)) - \nabla \cdot (\vec{u} \cdot \sigma) + \nabla \cdot q = \rho \vec{b} \tag{2.4.3}$$

For the modelling of some fluids, just the pressure force is taken into account. The Navier Stokes equations then reduce to the **Euler equations**:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{2.4.4}$$

$$\frac{\partial (\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u} + P) = 0 \tag{2.4.5}$$

$$\frac{\partial (\rho e_{tot})}{\partial t} + \nabla \cdot (\rho e_{tot} \vec{u} + P \vec{u}) = 0 \tag{2.4.6}$$

Both the Navier-Stokes and the Euler equations are highly under-determined. We have five equations describing the evolution of the conservative variables ρ, $\vec{u} = (u, v, w)$ and e_{tot}, but in addition to those we need information about the pressure P and the temperature T.
Material-dependent quantities like the thermal conductivity κ are also required, but they shall not be of our concern now. To read up on them, see for example [11].
We will consider gravity as our body force:

$$\vec{b} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} \tag{2.4.7}$$

g being the gravitational acceleration, $[g] = cm/s^2$

2.4.1 Ideal Gas

In ideal gases, the elementary particles are considered to be freely moving particles, moving in straight lines and once in a while colliding with other particles. The collision events are assumed to be perfectly elastic, and the particles are so small that they are many orders of magnitude smaller than the mean-free path between two collisions.
Obviously, this is an idealization of real gases, but if densities are low enough and temperatures are high enough, a gas typically starts to behave more and more as an ideal gas.

The ideal **equation of state (EOS)** links the temperature and the pressure of the gas particles:

$$P = \frac{\rho RT}{\mu} \tag{2.4.8}$$

where $R = 8.314472 \cdot 10^7 erg/(K \cdot mol)$ is the **gas constant** and μ denotes the **mean molecular weight** of the gas. Since we usually work with dimensionless quantities, $\mu = 1$ in case of a homogenous mixture.
Another aspect of an ideal gas is the equation of state relating the pressure to the internal specific energy e:

$$P = (\gamma - 1)\rho e \tag{2.4.9}$$

γ is the **adiabatic index** of the gas. A detailed derivation of γ is found in [2], p. 226. For our purposes we shall take

$$\gamma = \frac{5}{3}$$

For isothermal sound waves (in which e is constant), the sound speed is given by

$$c_s = \sqrt{\gamma \frac{P}{\rho}} \tag{2.4.10}$$

Chapter 3

Numerical Methods for Hyperbolic Conservation Laws

In this chapter we study numerical methods to solve systems of nonlinear, first-order hyperbolic partial differential equations, which arise as models of conservation laws.
Since the theory of numerical methods for hyperbolic systems is not as advanced as it is for scalar conservation laws we will concentrate on the description of numerical schemes and ignore issues of convergence theory.

3.1 Hyperbolic Conservation Laws

The following is taken from [3].

In the most general circumstance we would like to investigate a vector function

$$U(\mathtt{x},t) = (U^1(\mathtt{x},t),...,U^m(\mathtt{x},t)) \quad \mathtt{x} \in \mathbb{R}^n, t \geq 0$$

The components of U are the densities of various conserved quantities in some physical system.
Given then any smooth, bounded region $W \in \mathbb{R}^n$, we note that the integral

$$\int_W U(\mathtt{x},t)dV \tag{3.1.1}$$

represents the total amount of these quantities within W at time t.

Conservation laws typically assert that the rate of change within W is governed by a **flux function** $F : \mathbb{R} \to M_{m,n}(\mathbb{R})$, which controls the rate of loss or increase of U through ∂W.

Also, potential exterior body forces $B(\mathtt{x},t)$ have to be considered. So we arrive at

$$\frac{d}{dt}\int_W U(\mathtt{x},t)dV = -\int_{\partial W} F(U)\vec{n}dS + \int_W B(\mathtt{x},t) \tag{3.1.2}$$

$\vec{n}$ denoting the outward unit normal along W.

By Gauss's Divergence Theorem 1.2.1 we deduce

$$\frac{d}{dt}\int_W U(\mathtt{x},t)dV = -\int_W \nabla\cdot F(U)dV + \int_W B(\mathtt{x},t) \tag{3.1.3}$$

As the region W was arbitrary, we derive an initial-value problem for a general **system of conservation laws**:

$$\frac{\partial}{\partial t}U + \nabla\cdot F(U) = B \text{ in } \mathbb{R}\times(0,\infty) \tag{3.1.4}$$

$$U(\mathtt{x},0) = G(\mathtt{x}) \text{ on } \mathbb{R}\times\{t=0\} \tag{3.1.5}$$

the given function $G = (G^1,...,G^m)$ describing the initial distribution of $U = (U^1,...,U^m)$.

We will henceforth just consider on spatial dimension. The generalization to higher dimensions is easily done. We will denote the differentiation by subscripts:

$$U_t := \frac{\partial}{\partial t}U \tag{3.1.6}$$

$$(F(U))_x := \frac{\partial}{\partial x}F(U) \tag{3.1.7}$$

The initial-value problem (3.1.4) becomes

$$U_t + (F(U))_x = B \text{ in } \mathbb{R}\times(0,\infty) \tag{3.1.8}$$

$$U(\mathtt{x},0) = G(\mathtt{x}) \text{ on } \mathbb{R}\times\{t=0\} \tag{3.1.9}$$

Definition 3.1.1. Performing the spatial differentiation in equation (3.1.8) leads to

$$U_t + (DF)\cdot U_x = B$$

$DF = (\frac{\partial F^i}{\partial U^j})_{ij}$ is the Jacobian of $F(U)$. If all eigenvalues of DF are real, we call the system (3.1.8) **hyperbolic**.

Example: Euler's Equations

We recall Euler's equations from 2.4. In one dimension they read:

$$\rho_t + (\rho u)_x = 0 \tag{3.1.10}$$

$$(\rho u)_t + (\rho u^2 + p)_x = 0 \tag{3.1.11}$$

$$(\rho e_{tot})_t + ((\rho e_{tot} + p)u)_x = 0 \tag{3.1.12}$$

Writing $U(\mathtt{x}, t) = (\rho, \rho u, \rho e_{tot})^t$ and $F(U) = (\rho u, \rho u^2 + p, \rho e_{tot} + p)u)^t$ we check Euler's equations form a linear system of conservation laws.
The Jacobian DF of $F(U)$ is

$$DF = \begin{pmatrix} 0 & 1 & 0 \\ -u & 2u & 0 \\ -\frac{(\rho e_{tot}+p)u}{\rho} & \frac{(\rho e_{tot}+p)}{\rho} & u \end{pmatrix} \tag{3.1.13}$$

The eigenvalues of DF are $u + c, u,$ and $u - c$, c denoting the sound speed defined in section 2.4.1. Since they are all real, the system of conservation laws composed by Euler's equations is hyperbolic.

3.2 Basics of Discretization

This is taken from [14], [20].
In order to approximate the solution of the initial-value problem (3.1.8), we discretize space by a finite increasing sequence of grid points $a < x_1 < x_2 < ... < x_I = b$. We choose a constant spatial increment h and define

$$x_i = a + i \cdot h \qquad \text{i} = 0, \ldots \text{ n}$$
$$x_{i+\frac{1}{2}} = (i + \tfrac{1}{2}) \cdot h$$

Analogously we discretize time by a series of timepoints $0 < t_1 < t_2 < ... < t_N = T$ with a temporal increment τ

$$t^n = \tau \cdot n$$

We define the computational grid cells I_i to be the intervals $(x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}})$. Obviously, the **grid cell width** is h and a **timestep** is $\tau = t^n - t^{n-1}$. See Figure (3.1) for an illustration of spatial and temporal discretization.

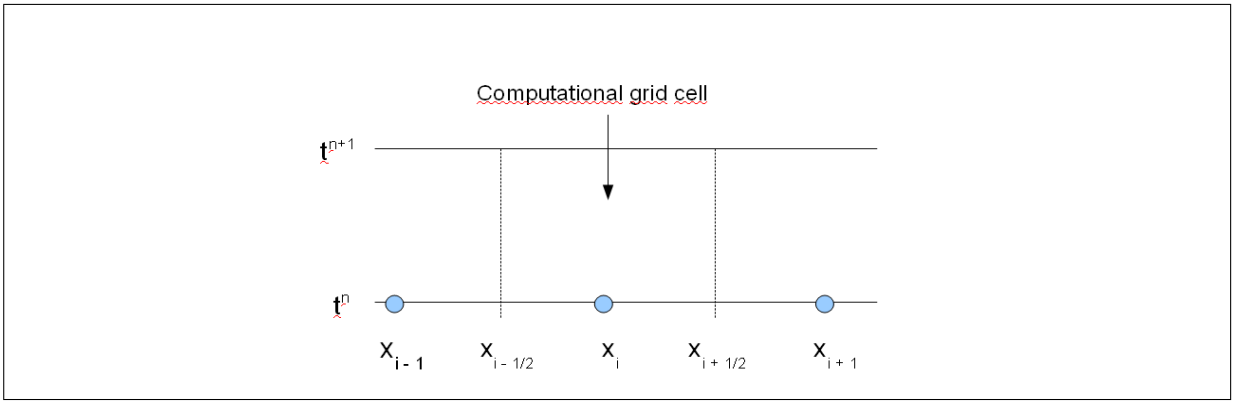


Figure 3.1: Spatial and temporal discretization

3.3 Conservative schemes

Let us consider a linear system of conservation laws:

$$U_t + F(U)_x = 0 \tag{3.3.1}$$

We integrate equation (3.3.1) over the space-time rectangle $(x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}) \times (t^n, t^{n+1})$ to find that for all $0 \leq i \leq I$ and $0 \leq n \leq N$

$$0 = \int_{t^n}^{t^{n+1}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (U_t + F(U)_x)\, dxdt =$$

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x,t)|_{t^n}^{t^{n+1}} dx + \int_{t^n}^{t^{n+1}} F(U)|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} dt =$$

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x,t^{n+1})dx - \int_{x_{j-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x,t^n)dx + \int_{t^n}^{t^{n+1}} F(U(x_{i+\frac{1}{2}},t))dt - \int_{t^n}^{t^{n+1}} F(U(x_{i-\frac{1}{2}},t))dt$$

This suggests how we construct numerical approximations to cell averages of the conserved quantities. Our numerical scheme will involve discrete quantities U_i^n that approximate the cell averages in the following sense:

$$\overline{U}_i^n \approx \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x,t^n)dx \tag{3.3.2}$$

Using this definition and dividing by h, equation (3.3.2) reads

$$\overline{U}_i^{n+1} = \overline{U}_i^n - \frac{1}{h}\left(\int_{t^n}^{t^{n+1}} F(U(x_{j+\frac{1}{2}},t))dt - \int_{t^n}^{t^{n+1}} F(U(x_{j-\frac{1}{2}},t))dt \right) \tag{3.3.3}$$

Similarly, we will work with a numerical flux function ϕ that approximates the time

averages of the flux function:

$$\phi\left(U_{i-p}^n, \cdots, U_{i+q}^n\right) \approx \frac{1}{\tau}\int_{t^n}^{t^{n+1}} F(U(x_{i+\frac{1}{2}}, t))dt \tag{3.3.4}$$

So equation (3.3.3) becomes

$$\overline{U}_i^{n+1} = \overline{U}_i^n - \frac{\tau}{h}\left(\phi\left(U_{i-p}^n, \cdots, U_{i+q}^n\right) - \phi\left(U_{i-p-1}^n, \cdots, U_{i+q-1}^n\right)\right) \tag{3.3.5}$$

Definition 3.3.1. Schemes of the form of (3.3.5) are called **conservative**.

Remark 3.3.2. Conservative finite difference schemes for solving the conservation law are distinguished solely by their choice for the numerical flux function ϕ.

For stability and consistency reasons, the temporal increment τ is restricted by the **Courant-Friedrichs-Levy (CFL) condition** for hyperbolic systems:

$$\max_{i=0,..,m}(|\lambda_i|)\cdot\tau \leq h \tag{3.3.6}$$

Once we find an accurate approximation for ϕ, an ordinary differential equation of the form $U_t = L(U)$ remains to be solved.

3.4 Essentially Non-Oscillatory (ENO) Schemes

ENO methods are highly nonlinear schemes developed to solve hyperbolic conservation laws, see [9], [17], [19]. The key idea is to interpolate the fluxes at cell boundaries using the smoothest stencil among several candidates to a high order accuracy and at the same time avoid spurious oscillations near shocks.
For systems of conservation laws, the approximation is done via upwinding in the eigensystem. There the equations decouple and the upwindig directions can be chosen correctly.

3.4.1 Transformation onto the Eigensystem

Let us consider a linear system of conservation laws as defined in (3.1.8),

$$U_t + F(U)_x = 0 \tag{3.4.1}$$

We calculate the Jacobian DF of $F(U)$:

$$DF = (\frac{\partial F^i(U)}{\partial(U^j)})_{ij} \tag{3.4.2}$$

Since we assume the system to be hyperbolic, there exist m real eigenvalues

$$\lambda_1(U) \leq \cdots \leq \lambda_m(U) \tag{3.4.3}$$

with corresponding left and right eigenvectors

$$l_1(U), l_2(U), ..., l_m(U) \tag{3.4.4}$$
$$r_1(U), r_2(U), ..., r_m(U) \tag{3.4.5}$$

which diagonalize the Jacobian DF.

We define

$$\Lambda(U) = diag(\lambda_i),\ \Lambda(U) \in M_n(\mathbb{R}$$
$$R(U) = (r_1(U), ..., r_m(U)),\ R(U) \in M_{n,m}(\mathbb{R})$$
$$L(U) = \begin{pmatrix} l_1(U) \\ \vdots \\ l_m(U) \end{pmatrix},\ L(U) \in M_{m,n}(\mathbb{R})$$

Basic linear algebra states that

$$F'(U)R(U) = R(U)\Lambda(U) \tag{3.4.6}$$

and

$$R^{-1}(U)F'(U)R(U) = \Lambda(U) = \begin{pmatrix} \lambda_1(U) & 0 & \cdots & 0 \\ 0 & \lambda_2(U) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m(U) \end{pmatrix} \tag{3.4.7}$$

We diagonalize the system near a fixed point x_0 using the transformations

$$\Lambda_0 = \Lambda(U(x_0)) \tag{3.4.8}$$
$$L_0 = L(U(x_0)) \tag{3.4.9}$$
$$R_0 = R(U(x_0)) \tag{3.4.10}$$

Since these transformations are constant, we can move them inside the derivatives:

$$\begin{aligned}(L_0U)_t &= -(L_0F(U))_x \\ &= -(L_0F'(U)R_0)(L_0U_x) \\ &= -\Lambda_0(L_0U)_x\end{aligned}$$

We see that the characteristic variables decouple. The scalar components of this system are upwinded separately and with use of R_0 the results are transformed back to the physical system:

$$U_t = -(R_0\Delta(L_0F(U)))_x \tag{3.4.11}$$

Δ denotes the upwind discretization operator.

3.4.2 Reconstruction and Approximation in 1D

Be $v(x)$ a function given on the grid points x_i by $v_i := v(x_i)$. We are looking for a numerical flux function $\hat{v}_{i+\frac{1}{2}} = \hat{v}(v_{i-r}, ..., v_{i+s})$ such that the difference of two values of this numerical flux function approximates the derivative $v'(x)$ to k^{th} order accuracy:

$$\frac{1}{h}(\hat{v}_{i+\frac{1}{2}} - \hat{v}_{i-\frac{1}{2}}) = v'(x_i) + O(h^k) \tag{3.4.12}$$

We want to find a function $g(x)$ that satifies

$$v(x) = \frac{1}{h}\int_{x-\frac{h}{2}}^{x+\frac{h}{2}} g(\xi)d\xi \tag{3.4.13}$$

because then

$$v'(x) = \frac{1}{h}(g(x+\frac{h}{2}) - g(x-\frac{h}{2})) \tag{3.4.14}$$

holds.

Note that $g(x)$ in equation (3.4.13) is defined solely by its cell averages. To reconstruct $g(x)$ we need the following algorithm:

Reconstruction from cell averages

We are given cell averages $\overline{v}_i$ of a function $v(x)$,

$$\overline{v}_i = \frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} v(\xi)d\xi \tag{3.4.15}$$

For every computational grid cell I_i we look for a polynomial $p_i(x)$ of order $\leq k-1$ that approximates $v(x)$ in the interior of I_i:

$$p_i(x) = v(x) + O(h^k) \tag{3.4.16}$$

Given the location I_i and the order of accuracy k, we choose a fixed stencil S_i based on r cells to the left, s cells to the right and I_i itself:

$$S(i) = (I_{i-r}, ..., I_{i+s})$$
$$r + s + 1 = k \qquad 0 \leq r, s$$

Basic interpolation theory states that there is a unique polynomial $p_i(x)$ of order at most $k-1 = r+s$, whose cell average in each of the cells in $S(i)$ agrees with that of $v(x)$:

$$\frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} p(\xi)d\xi = \overline{v}_i \tag{3.4.17}$$

We claim that $p_i(x)$ and further satisfies equation (3.4.16). To verify this and to construct $p_i(x)$ we define the primitive function of $v(x)$:

$$V(x) := \int_{-\infty}^{x} v(\xi)d\xi \tag{3.4.18}$$

V(x) can be expressed by the cell averages $\overline{v}_i$:

$$V(x_{i+\frac{1}{2}}) = \sum_{j=-\infty}^{i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} v(\xi)d\xi = \sum_{j=-\infty}^{i} \overline{v}_j h \tag{3.4.19}$$

Thus the knowledge of the cell averages gives exact knowledge of $V(x)$ at the cell boundaries. Be $P(x)$ the unique polynomial of degree $\leq k$ which interpolates $V(x_{i+\frac{1}{2}})$ at the

$(k+1)$-points $V(x_{i-r-\frac{1}{2}}), ..., V(x_{i+s+\frac{1}{2}})$ and denote its derivative by $p(x)$. We deduce

$$\begin{aligned}
\frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} p(\xi)d\xi = \frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} P(\xi)d\xi = \\
\frac{1}{h}(P(x_{i+\frac{1}{2}}) - P(x_{i-\frac{1}{2}})) = \\
\frac{1}{h}(V(x_{i+\frac{1}{2}}) - V(x_{i-\frac{1}{2}})) = \\
\frac{1}{h}\left(\int_{-\infty}^{x_{i+\frac{1}{2}}} v(\xi)d\xi - \int_{-\infty}^{x_{i-\frac{1}{2}}} v(\xi)d\xi\right) = \\
\frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} v(\xi)d\xi = \overline{v}_i
\end{aligned}$$

Standard apprioximation theory (see [16]) yields

$$P'(x) = V'(x) + O(h^k)$$

This is the accuracy proposed in equation (3.4.16).

At cell boundaries we get k$^{\text{th}}$ order accurate approximations of $v(x)$:

$$\begin{aligned}
v^-_{x_{i+\frac{1}{2}}} &= p_i(x_{i+\frac{1}{2}}) = v(x_{i+\frac{1}{2}}) + O(h^k) \\
v^+_{x_{i-\frac{1}{2}}} &= p_i(x_{i-\frac{1}{2}}) = v(x_{i-\frac{1}{2}}) + O(h^k)
\end{aligned}$$

Since the mappings from the given cell averages v_i in the stencil $S(i)$ to the values at the cell boundaries are linear, there exist constants $c_{r,j}$ and $\hat{c}_{r,j}$ depending on the left shift r of the stencil and the accuracy k, but not on the function values v_i itself so that

$$\begin{aligned}
v^-_{x_{i+\frac{1}{2}}} &= \sum_{j=0}^{k-1} c_{r,j}\overline{v}_{i-r+j} \\
v^+_{x_{i+\frac{1}{2}}} &= \sum_{j=0}^{k-1} \hat{c}_{r,j}\overline{v}_{i-r+j}
\end{aligned}$$

The superscripts $+$ and $-$ are due to the possibility of different stencils for cell I_i and I_{i+1} when interpolating to location $x_{i+\frac{1}{2}}$. If the left shift r is not identified with cell I_i but with the point of reconstruction $x_{i+\frac{1}{2}}$ we get

$$c_{r-1,j} = \hat{c}_{r,j} \tag{3.4.20}$$

Thus we have found a numerical flux function $\hat{v}_{i+\frac{1}{2}} = \sum_{j=0}^{k-1} c_{rj} v_{i-r+j}$ satisfying equation (3.4.14). The constants c_{rj} are derived in [19].

3.4.3 Weighted Essentially Non-Oscillatory (WENO) Schemes

Standard ENO-methods use a fixed stencil for every point x_i. However, if there is a discontinuity is encountered between the points x_i and x_{i+1}, the fixed stencil approximation can lead to overshooting.
The idea of a WENO scheme is to use a convex combination of all possible candidate stencils instead of choosing only one. Each of those stencils is assigned a weight which determines the contribution to the final approximation of the numerical flux. In smooth regions, the weights can be defined such that the scheme it approaches certain optimal weights to achieve a higher order of accuracy - a r^{th} order ENO scheme leads to a $(2r-1)^{th}$ order WENO scheme in the optimal case - while in regions near discontinuities, the stencils which contain the discontinuities are assigned a nearly zero weight.
This is taken from [9], [17].

Interpolating on all k candidate stencils $S_r(i) = \{I_{i-r}, ..., I_{i-r+k-1}\}$ produces k different reconstructions to the value $\hat{v}_{i+\frac{1}{2}}$,

$$\hat{v}_{i+\frac{1}{2}}^{(r)} = \sum_{j=0}^{k-1} c_{rj} \overline{v}_{i-r+j} \tag{3.4.21}$$

The new WENO approximation takes

$$\hat{v}_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} \omega_r \hat{v}_{i+\frac{1}{2}}^{(r)} \tag{3.4.22}$$

where the weights ω_r are required to fullfill

$$\omega_r \geq 0 \tag{3.4.23}$$

$$\sum_{r=0}^{k-1} \omega_r = 1 \tag{3.4.24}$$

for reasons of stability and consistency.
If the function $v(x)$ is smooth in all of the candidate stencils, there exist constants d_r such that

$$\hat{v}_{i+\frac{1}{2}} = \sum_{r=0}^{k-1} d_r \hat{v}_{i+\frac{1}{2}}^{(r)} = v(x_{i+\frac{1}{2}}) + O(h^{2k-1}) \tag{3.4.25}$$

and

$$\sum_{r=0}^{k-1} d_r = 1 \tag{3.4.26}$$

In [19] these constants are denoted as optimal weights.

If

$$\omega_r = d_r + O(h^{k-1})$$

holds, $\hat{v}_{i+\frac{1}{2}}$ is (2k-1)$^{\text{th}}$ order accurate:

$$\sum_{r=0}^{k-1} \omega_r \hat{v}_{i+\frac{1}{2}}^{(r)} - \sum_{r=0}^{k-1} d_r \hat{v}_{i+\frac{1}{2}}^{(r)} = \tag{3.4.27}$$

$$\sum_{r=0}^{k-1} (\omega_r - d_r)(\hat{v}_{i+\frac{1}{2}}^{(r)} - v(x_{i+\frac{1}{2}}) = \tag{3.4.28}$$

$$\sum_{r=0}^{k-1} O(h^{k-1}) O(h^k) = O(h^{2k-1}) \tag{3.4.29}$$

In case of a discontinuity of the function $v(x)$ in one or more of the stencils, we expect the corresponding weight(s) to be essentially zero. In [19] the weights are defined

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{k-1} \alpha_s} \tag{3.4.30}$$

with

$$\alpha_r = \frac{d_r}{(\epsilon + \beta_r)^2} \tag{3.4.31}$$

ϵ is a positive real number, introduced to prevent the denominator becoming zero, whereas β_r is a smoothness measurement on the r$^{\text{th}}$ stencil:

$$\beta_r = \sum_{l=0}^{k-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} h^{2l-1} (p_r^{(l)}(\xi))^2 d\xi \tag{3.4.32}$$

and $p_r^{(l)}$ denotes the l$^{\text{th}}$ derivative of the interpolating polynomial $p_r(x)$.

Constants for k=3

For 5$^{\text{th}}$ order WENO, we have $k = 3$. The corresponding constants c_{rj} are listed below:

r	j=1	j=2	j=3
-1	11/6	-7/6	1/3
0	1/3	5/6	-1/6
1	-1/6	5/6	1/3
2	1/3	-7/6	11/6

The optimal weights d_r are given by

	r=0	r=1	r=2
d	3/10	3/5	1/10

and the smoothness indicators β_r are calculated as

$$\beta_0 = \frac{13}{12}(\overline{v}_i - 2\overline{v}_{i+1} + \overline{v}_{i+2})^2 + \frac{1}{4}(3\overline{v}_i - 4\overline{v}_{i+1} + \overline{v}_{i+2})^2$$
$$\beta_1 = \frac{13}{12}(\overline{v}_{i-1} - 2\overline{v}_i + \overline{v}_{i+1})^2 + \frac{1}{4}(\overline{v}_{i-1} - \overline{v}_{i+1})^2$$
$$\beta_2 = \frac{13}{12}(\overline{v}_{i-2} - 2\overline{v}_{i-1} + \overline{v}_i)^2 + \frac{1}{4}(\overline{v}_{i-2} - 4\overline{v}_{i-1} + 3\overline{v}_i)^2$$

3.5 The Complementary Projection Method

The standard construction of WENO schemes requires finding the Jacobian matrix of the flux function and the associated eigensystem.
However, there are also systems of conservation laws which have multiple eigenvalues. In such a system, the distinct eigenvalues have corresponding unique eigenspaces. However, the eigenvectors for the manifold eigenvalues are not unique, so to obtain a complete eigensystem they must be chosen arbitrarily.
Based on the WENO scheme presented in section 3.4, the Complementary Projection Method of [18] avoids this arbitrary choice of eigenvectors.

Let us again consider our hyperbolic system of conservation laws:

$$U_t + F(U)_x = 0 \tag{3.5.1}$$

Be l^i and r^i the left and right eigenvectors of the Jacobian DF of the flux function, with associated eigenvalues λ^i. The left and right eigenvectors are further required to be mutually orthonormal,

$$l^i \cdot r^i = \delta_{ij} \tag{3.5.2}$$

δ_{ij} denotes the Kronecker delta.

Given this complete eigensystem, any upwind difference scheme defined for scalar equations can be extended to the hyperbolic system via a characteristic decomposition, i.e. transformation into the eigensystem where the equations decouple. The spatial discretization of $F(U)_x$ is then expressed as the difference of fluxes between two grid cell walls.

To define the flux at a particular cell wall, we transform the vector fluxes at each cell center into scalar fluxes for the i$^{\text{th}}$ characteristic field by

$$f^i = l^i_w \cdot F(U) \tag{3.5.3}$$

l^i_w, r^i_w and λ^i_w denote left and right eigenvectors and eigenvalues evaluated at the wall in some fashion.

The assumed orthonormality implies

$$F(U) = f^1 r^1_w + f^2 r^2_w + ... + f^n r^n_w \tag{3.5.4}$$

Next, for each scalar field i the cell center characteristic fluxes, f^i, are interpolated to the cell wall of interest in an upwind fashion. Thus we obtain the scalar characteristic wall flux f^i_w.

Finally, the desired total wall flux vector is defined as the sum of all the characteristic vector contributions,

$$F(U)_w = f^1_w r^1_w + f^2_w r^2_w + ... + f^n_w r^n_w \tag{3.5.5}$$

To introduce now the alternative approach, let us suppose that from the n eigenvalues we have an eigenvalue with multiplicity p. Without loss of generality, we assume

$$\lambda^1_w = \lambda^2_w = ... = \lambda^p_w$$

The corresponding subspace is the span of the eigenvectors, $l^1_w, ...l^p_w$ and the part of the original cell center flux vector that lies in this subspace is

$$\mathfrak{F} = f^1 r^1_w + f^2 r^2_w + ... + f^p r^p_w \tag{3.5.6}$$

All of the characteristic fields contributing to $\mathfrak{F}$ have the same upwind direction, since their eigenvalues are identical. So $\mathfrak{F}$ has a well-defined upwind direction and upwind differencing is possible without decomposing $\mathfrak{F}$ further into scalar fluxes. Therefore, we apply dirctly upwind interpolation to the cell center values of the vector $\mathfrak{F}$ in a component by component

fashion. Let F_w denote the resulting flux value, then we can rewrite equation (3.5.5) to

$$F(U)_w = \mathfrak{F}(U)_w + f_w^{p+1} r_w^{p+1} + ... + f_w^n r_w^n \tag{3.5.7}$$

So we can compute $\mathfrak{F}$ without knowing the basis of left and right eigenvectors defining it,

$$F = \mathfrak{F}(U) - (f^{p+1} r_w^{p+1} + ... + f^n r_w^n) \tag{3.5.8}$$

In practice we define the cell wall flux via equation (3.5.7) and compute F_w as calculated in equation (3.5.8), which requires only the left and right eigenvectors associated with $\lambda_w^{p+1}, ...\lambda_w^n$.

Properties of the CPM

(i) Computational Cost
When applied to a system with a repeated eigenvalue having a large multiplicity, the complementary projection method may require fewer operations and therefore result in a faster code.

(ii) Distinct Multiple Eigenvalues
This technique can only be used to define a single basis-free projection: We can project onto a subspace S_1 without a basis for it, given a basis for its complement. But if we need projection operators for two linearly independent subspaces S_1 and S_2, i.e. if the eigensystem of a flux function $F(U)$ has two distinct repeated eigenvalues, we must select an eigenbasis for at least one of them. Then the other can be treated without a basis.

3.6 Runge-Kutta (RK) Methods

Since we have found a possibility to approximate the divergence term $F(U)_x$ in our system of conservation laws, it reduces to a system of ordinary differential equations of the general appearance

$$U_t = L(U) \tag{3.6.1}$$

For the numerical solutions of initial-value problems of this kind, there are many methods available. In this section we introduce Runge-Kutta schemes.

Runge-Kutta methods are one-step methods, meaning that they step forward from computed approximations $U^{(n)}$ at times t^n to new approximations $U^{(n+1)}$ using only $U^{(n)}$ as

input. During a step, however, auxiliary intermediate approximations serve to obtain a sufficiently high accuracy at the main step points $U^{(n)}$.

A general Runge-Kutta scheme is written in the form

$$U^{(i)} = \sum_{k=0}^{i-1} (\alpha_{ik} U^{(k)} + \tau \beta_{ik} L(U^{(k)}),\ i = 1, ..., m \tag{3.6.2}$$

$$U^{(0)} = U^{(n)} \tag{3.6.3}$$

$$U^{(m)} = U^{(n+1)} \tag{3.6.4}$$

For consistency reasons, we require

$$\sum_{k=0}^{i-1} \alpha_{ik} = 1 \tag{3.6.5}$$

Example: First order Euler-forward timestepping

If $m = 1$ then equation (3.6.5) leads to $\alpha_{10} = 1$. Be $\beta_{10} = 1$. The resulting Runge-Kutta scheme

$$U^{(n+1)} = U^{(n)} + \tau L(U^{(n)}) \tag{3.6.6}$$

is exactly the first order Euler-forward scheme.

Remark 3.6.1. Clearly, if all the coefficients are nonnegative $\alpha_{ik} \geq 0$, $\beta_{ik} \geq 0$, the Runge-Kutta method is a convex combination of Euler-forward operators.

In [4] the optimal second order Runge-Kutta method is given by

$$U^{(1)} = U^{(n)} + \tau L(U^{(n)}) \tag{3.6.7}$$

$$U^{(n+1)} = \frac{1}{2} U^{(n)} + \frac{1}{2} U^{(1)} + \frac{1}{2} \tau L(U^{(1)}) \tag{3.6.8}$$

and the third order Runge-Kutta scheme is

$$U^{(1)} = U^{(n)} + \tau L(U^{(n)}) \tag{3.6.9}$$

$$U^{(2)} = \frac{3}{4} U^{(n)} + \frac{1}{4} U^{(1)} + \frac{1}{4} \tau L(U^{(1)}) \tag{3.6.10}$$

$$U^{(n+1)} = \frac{1}{3} U^{(n)} + \frac{2}{3} U^{(2)} + \frac{2}{3} \tau L(U^{(2)}) \tag{3.6.11}$$

$$\tag{3.6.12}$$

Chapter 4

A Fractional Step Method for the Euler Equations

Time integration of the equations of hydrodynamics is often carried out by means of a **fractional step procedure** whereby the momentum equations and some form of Poisson equation are solved separately at each time step. The Poisson equation is constructed from the equations of mass and momentum and acts to enforce continuity.
N. Kwatra et alii proposed in [10] such a scheme with the objective to alleviate the stringent CFL-condition imposed by the sound speed in simulating inviscid compressible flows with shocks, contacts and rarefactions. The general idea is to filter the sound waves, which transport a negligible amount of energy but travel very fast, so to resolve them, small time steps have to be chosen.

Let us consider the one dimensional Euler equations,

$$\begin{pmatrix} \rho \\ \rho u \\ \rho e_{tot} \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho e_{tot} u + P u \end{pmatrix}_x = 0 \tag{4.0.1}$$

We separate the flux term into an advection part $F_1(U)$ and a non-advection part $F_2(U)$:

$$F_1(U) = \begin{pmatrix} \rho u \\ \rho u^2 \\ \rho e_{tot} u \end{pmatrix} \tag{4.0.2}$$

$$F_2(U) = \begin{pmatrix} 0 \\ P \\ P u \end{pmatrix} \tag{4.0.3}$$

The Jacobian DF_1 of the advection part is

$$DF_1 = \begin{pmatrix} 0 & 1 & 0 \\ -u^2 & 2u & 0 \\ -\frac{\rho e_{tot} u}{\rho} & \frac{\rho e_{tot}}{\rho} & u \end{pmatrix} \tag{4.0.4}$$

Computing the eivenvalues of DF_1, we find that they are all identical to u and DF_1 is rank deficient.

We first update the advection part, by applying the Complementary Projection Method of section 3.5. Since we have just a 3-fold (in higher dimensions 4- and 5-fold) eigenvalue, we do not have to transform onto the characteristic variables whatsoever.

Thus we obtain intermediate values of the conserved variables $\rho^*, (\rho\vec{u})^*$ and $(\rho e_{tot})^*$. Since the pressure does not affect the continuity equation, $\rho^* = \rho^{n+1}$.

Now we need to correct these values to time t^{n+1}. The momentum and energy updates are

$$\frac{(\rho\vec{u})^{n+1} - (\rho\vec{u})^*}{\tau} = -\nabla P^{n+1} \tag{4.0.5}$$

and

$$\frac{(\rho e_{tot})^{n+1} - (\rho e_{tot})^*}{\tau} = -\nabla \cdot (P\vec{u})^{n+1} \tag{4.0.6}$$

We divide (4.0.5) by ρ^{n+1}

$$\vec{u}^{n+1} = \vec{u}^* - \tau \frac{\nabla P^{n+1}}{\rho^{n+1}} \tag{4.0.7}$$

and take its divergence to obtain

$$\nabla \cdot \vec{u}^{n+1} = \nabla \cdot \vec{u}^* - \tau \nabla \cdot \frac{\nabla P^{n+1}}{\rho^{n+1}} \tag{4.0.8}$$

Now we use a pressure evolution equation $\frac{\partial P}{\partial t} + \vec{u} \cdot \nabla P = -\rho c^2 \nabla \cdot \vec{u}$ to predict the pressure at time $n+1$. It is derived in section 4.1.

We fix $\vec{u}$ to be at time $n+1$ through the time step and substitute in equation (4.0.8) to get

$$\frac{\partial P}{\partial t} + \vec{u} \cdot \nabla P = -\rho c_s^2 \nabla \cdot \vec{u} + \rho c_s^2 \tau \nabla \cdot \frac{\nabla P}{\rho^{n+1}} \tag{4.0.9}$$

This 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 in $\rho c^2 \tau \nabla \cdot \frac{\nabla P}{\rho^{n+1}}$ at time t^{n+1} as is typical for backward Euler discretization gives after rearrangement

$$P^{n+1} - \rho^n (c_s^2)^n \tau^2 \nabla \cdot \frac{\nabla P^{n+1}}{\rho^{n+1}} = P^n - (\vec{u}^n \cdot \nabla P^n)\tau - \rho^n (c_s^2)^n \tau \nabla \cdot \vec{u}^* \tag{4.0.10}$$

c_s denotes the sound speed.
This equation can be further simplified by using an advection equation for pressure,

$$\frac{P^a - P^n}{\tau} + \vec{u} \cdot \nabla P^n = 0 \tag{4.0.11}$$

Rearranging gives

$$P^a = P^n - (\vec{u}^n \cdot \nabla P^n)\tau \tag{4.0.12}$$

and substituting in equation (4.0.10) leads to

$$P^{n+1} - \rho^n (c_s^2)^n \tau^2 \nabla \cdot \frac{\nabla P^{n+1}}{\rho^{n+1}} = P^a - \rho^n (c_s^2)^n \tau \nabla \cdot \vec{u}^* \tag{4.0.13}$$

Updating the pressure

Equation (4.0.13) is a general elliptic partial differential equation of the form

$$cP + \nabla(\kappa \nabla P) = -f \tag{4.0.14}$$

with

$$c = \frac{1}{\rho^n (c_s^2)^n \tau^2} \tag{4.0.15}$$

$$\kappa = \frac{1}{\rho^{n+1}} \tag{4.0.16}$$

$$f = -(\frac{1}{\rho^n (c_s^2)^n \tau^2} P_a + \frac{1}{\tau} \nabla \cdot \vec{u}^*) \tag{4.0.17}$$

We discretize this equation at cell centers. To compute $\vec{u}^*$ in f we need to define velocities at cell faces. Consider two adjacent grid cells, one centered at x_i and one centered at x_{i+1}, as shown in Figure 4.1.
We divide these into four regions $C_{i,L}, C_{i,R}, C_{i+1,L}, C_{i+1,R}$. Let h_x be the spatial increment in x-direction. Thus equation (4.0.5) for $C_{i,R}$ reads

$$\frac{(\rho\vec{u})_{i,R}^{n+1} - (\rho\vec{u})_{i,R}^*}{\tau} = -\frac{P_{i+\frac{1}{2}}^{n+1} - P_i^{n+1}}{\frac{h_x}{2}} \tag{4.0.18}$$

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

$$\frac{(\rho\vec{u})_{i,L}^{n+1} - (\rho\vec{u})_{i,L}^*}{\tau} = -\frac{P_{i+1}^{n+1} - P_{i+\frac{1}{2}}^{n+1}}{\frac{h_x}{2}} \tag{4.0.19}$$

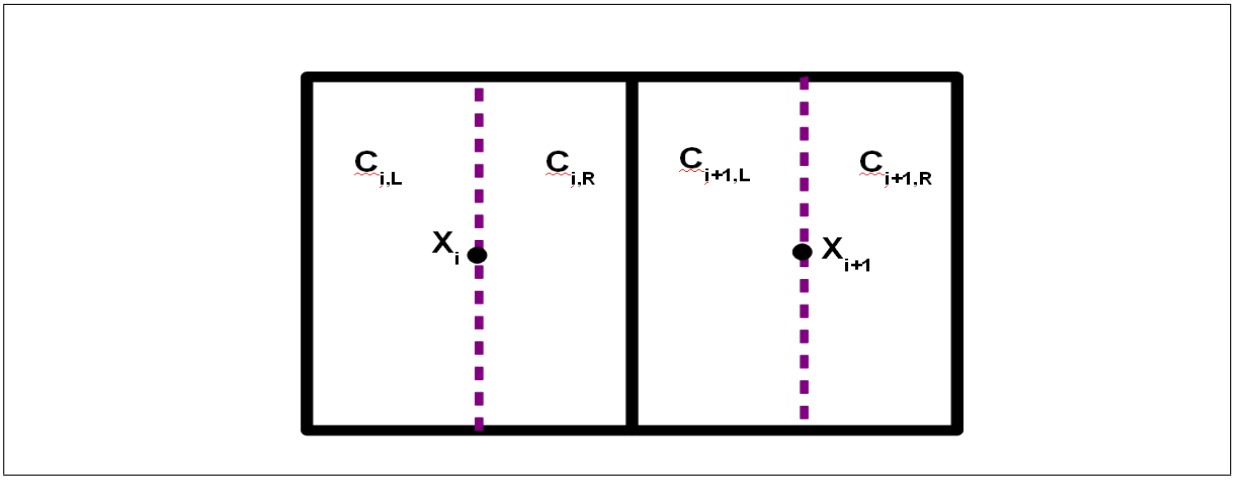


Figure 4.1: A dual cell

Adding equations (4.0.18) and (4.0.19) and dividing by $\rho_i^{n+1} + \rho_{i+1}^{n+1}$ yields

$$\frac{\hat{u}_{i+\frac{1}{2}}^{n+1} - \hat{u}_{i+\frac{1}{2}}^{*}}{\tau} = \frac{P_{i+1}^{n+1} - P_i^{n+1}}{h_x \hat{\rho}^{n+1}} \tag{4.0.20}$$

where

$$\hat{u}_{i+\frac{1}{2}} = \frac{(\rho u)_{i,R} + (\rho u)_{i+1,L}}{\rho_i + \rho_{i+1}} \tag{4.0.21}$$

can be thought of a density-weighted face velocity. 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. So equation (4.0.21) becomes

$$\hat{u}_{i+\frac{1}{2}} = \frac{(\rho u)_i + (\rho u)_{i+1}}{\rho_i + \rho_{i+1}} \tag{4.0.22}$$

The cell face density is calculated as

$$\hat{\rho}_{i+\frac{1}{2}} = \frac{\rho_{i+1} + \rho_i}{2} \tag{4.0.23}$$

Now we can calculate p_a, c and f to substitute into (4.0.13). This partial differential equation is solved numerically with use of a finite element scheme. See [5] for more details.

Thus we obtain P^{n+1} at cell centers.

Updating momentum and energy

To obtain the correct shock speeds we use a flux based method and therefore need the pressure at cell faces for equation (4.0.5) and (4.0.6). Applying conservation of momentum

to our control volumes $C_{i,R}$ and $C_{i+1,L}$ gives

$$\frac{Du_{i,R}}{Dt} = \frac{P_i - P_{i+\frac{1}{2}}}{\frac{h_x \rho_{i,R}}{2}} \tag{4.0.24}$$

and

$$\frac{Du_{i+1,L}}{Dt} = \frac{P_{i+\frac{1}{2}} - P_{i+1}}{\frac{h_x \rho_{i+1,L}}{2}} \tag{4.0.25}$$

The constraint that the interface remains in contact implies that

$$\frac{Du_{i,R}}{Dt} = \frac{Du_{i+1,L}}{Dt} \tag{4.0.26}$$

so we get

$$P_{i+\frac{1}{2}} = \frac{P_{i+1}\rho_i + P_i \rho_{i+1}}{\rho_{i+1} + \rho_i} \tag{4.0.27}$$

The cell face velocity is computed via equation (4.0.20) and $P_{i+\frac{1}{2}} \hat{u}_{i+\frac{1}{2}}$ is used in equation (4.0.5).

4.1 The Pressure Evolution Equation

In this section we derive an equation describing the pressure evolving in time. We construct it from the continuity and the momentum equation in (4.0.1).

Remark 4.1.1. We will denote the partial derivatives with subscripts:

$$\begin{aligned} \frac{\partial \rho}{\partial t} &:= \rho_t \\ \frac{\partial \rho}{\partial x} &:= \rho_x \\ &\vdots \end{aligned}$$

Consider a general equation of state,

$$P = P(\rho, e) \tag{4.1.1}$$

and a general definition for the **sound speed** c_s,

$$c_s^2 = P\rho + \frac{PP_e}{\rho^2} \tag{4.1.2}$$

As in Chapter 2, e denotes the specific internal energy, whereas $e_{tot} = e + \frac{u^2}{2}$ denotes the total energy.

First, we take the total temporal derivative of (4.1.1),

$$P_t = P_\rho \rho_t + P_e e_t \tag{4.1.3}$$

We remember the continuity and the momentum equation:

$$\rho_t + \rho_x u + \rho u_x = 0 \tag{4.1.4}$$

$$\rho_t u + \rho u_t + \rho_x u^2 + 2\rho u u_x + P_x = 0 \tag{4.1.5}$$

Substituting 4.1.4 into 4.1.5 yields

$$u_t + \rho u u_x = -P_x \tag{4.1.6}$$

Taking the temporal and spatial derivative of $\rho e_{tot} = \rho e + \frac{\rho u^2}{2}$,

$$(\rho e_{tot})_t = \rho_t e + e_t \rho + \frac{\rho_t u^2}{2} + \rho u u_t \tag{4.1.7}$$

$$(\rho e_{tot})_x = \rho_x e + \rho e_x + \frac{\rho_x u^2}{2} + \rho u u_x \tag{4.1.8}$$

and substituting this into (4.1.4) and (4.1.5) leads to

$$e_t + u e_x = -\frac{P}{\rho} u_x \tag{4.1.9}$$

We insert equation (4.1.9) in (4.1.3) and use (4.1.1):

$$P_t + u P_x = -\rho u_x (P_\rho + \frac{P P_e}{\rho^2}) \tag{4.1.10}$$

Using the definition of the sound speed in (4.1.2) we get

$$P_t + u P_x = -\rho c_s^2 u_x \tag{4.1.11}$$

This is the pressure evolution equation we use in our fractional step method to predict the pressure at the next time step.

Remark 4.1.2. In higher dimensions this becomes

$$P_t + u P_x + v P_y + w P_z = -\rho c_s^2 (u_x + v_y + w_z) \tag{4.1.12}$$

which is equivalent to

$$\frac{\partial P}{\partial t} + \vec{u} \cdot \nabla P = -\rho c_s^2 \nabla \cdot \vec{u} \tag{4.1.13}$$

The Algorithm

- Calculate the time t^n pressure P^n using an equation of state.
- Calculate $F_1(U)$ and apply the WENO algorithm to it. Since all eigenvalues of the Jacobian of $F_1(U)$ are equal, we use the Complementary Projection Method. This yields intermediate values for the conservative variables: $\rho^*, (\rho u)^*$ and $(\rho e_{tot})^*$.
- Calculate the pressure at time t^{n+1}:
  (i) Calculate the intermediate values of $\vec{u}$ at cell faces $\vec{u}^*_{i+\frac{1}{2}}$.
  (ii) Calculate c, κ and f for the Poisson-like pressure equation (4.0.13) and solve it.
- Update momentum and energy to time t^{n+1}:
  (i) Calculate $P^{n+1}_{i+\frac{1}{2}}$ via equation (4.0.27).
  (ii) Update $\vec{u}^*_{i+\frac{1}{2}}$ to $\vec{u}^{n+1}_{i+\frac{1}{2}}$ via equation (4.0.20).
  (iii) Update $(\rho u)^*$ and $(\rho e_{tot})^*$ to time t^{n+1} by equation (4.0.5) and (4.0.6).

4.2 MAC-ENO scheme

Using traditional ENO schemes for the advection part of the Euler equations lead to excessive spurious oscillations. Kwatra et alii. suggest in [10] that this might be related to the dual cell center formulation and have devised a new ENO-scheme which better utilizes this formulation. The main idea is to replace the advection velocity in the fluxes with the cell-face value $\vec{u}_{i+\frac{1}{2}}$ when constructing the numerical flux at location $x_{i+\frac{1}{2}}$. So instead of using $\rho_j\vec{u}_j, \rho_j\vec{u}_j^2$ and $\rho_j(e_{tot})_j$ we use $\vec{u}_j$, $\rho_j\vec{u}_{i+\frac{1}{2}}, \rho_j\vec{u}\vec{u}_{i+\frac{1}{2}}$ and $\rho_j(e_{tot})_j\vec{u}_{i+\frac{1}{2}}$
This MAC-ENO scheme (MENO) especially proves useful in suppressing two-point instabilities.

4.3 Time Step Restriction

Since we solve the acoustic component implicitly, we no longer have a severe time step restriction determined by the sound speed c_s. All that remains is to find an estimate for the maximum value of $|u|$ throughout the timestep. Since our tests in section 4.4 start out with an initial velocity identically zero, it is not enough to use u^n, since $u^n = 0$ would imply an infinite τ.

Assuming the flow is smooth, we combine conservation of mass and momentum to obtain an equation describing the velocity u evolving in time:
The continuity equation reads

$$0 = \rho_t + (\rho u)_x = \rho_t + \rho_x u + \rho u_x \tag{4.3.1}$$

and conservation of momentum becomes

$$0 = (\rho u)_t + (\rho u^2 + P)_x = \tag{4.3.2}$$

$$\rho_t u + \rho u_t + \rho_x u^2 + 2\rho u u_x + P_x \tag{4.3.3}$$

Inserting equation (4.3.1) into equation (4.3.3) leads to

$$u_t + u u_x + \frac{P_x}{\rho} = 0 \tag{4.3.4}$$

In one spatial dimension, we use this to estimate the velocity at the end of the timestep as

$$\frac{|u^n|_{max} + \frac{|P|_x}{\rho}\tau}{h_x} \tag{4.3.5}$$

Thus the CFL-condition becomes

$$\tau \left(\frac{|u^n|_{max} + \frac{|P|_x}{\rho}\tau}{h_x} \right) \leq 1 \tag{4.3.6}$$

This is quadratic in τ with solutions

$$\frac{-|u^n|_{max} - \sqrt{|u^n|^2_{max} + 4\frac{|P|_x}{\rho}h_x}}{\frac{2|P|_x}{\rho}} \leq \tau \leq \frac{-|u^n|_{max} + \sqrt{|u^n|^2_{max} + 4\frac{|P|_x}{\rho}h_x}}{\frac{2|P|_x}{\rho}} \tag{4.3.7}$$

Since the lower limit is always non positive, we only need to enforce the upperbound. The CFL condition now reads

$$\frac{\tau}{2}\left(\frac{|u^n|_{max}}{h_x} + \sqrt{\left(\frac{|u^n|_{max}}{h_x}\right)^2 + 4\frac{|P|_x}{\rho h_x}} \right) \leq 1 \tag{4.3.8}$$

and in two spatial dimensions equation (4.3.8) becomes

$$\frac{\tau}{2}\left(\frac{|u|_{max}}{h_x} + \frac{|v|_{max}}{h_y} + \sqrt{\left(\frac{|u|_{max}}{h_x} + \frac{|v|_{max}}{h_y}\right)^2 + 4\frac{|P|_x}{\rho h_x} + 4\frac{|P|_y}{\rho h_y}} \right) \leq 1 \tag{4.3.9}$$

4.4 Numerical results

All of our simulations are carried out using fifth order WENO and a second order Runge-Kutta scheme. Kwatra et alii suggest in [10] two variations of the RK scheme: The first is to perform RK on just the advection part, $F_1(U)$, with only one final implicit solve for $F_2(U)$. The second variation is to carry out both $F_1(U)$ and $F_2(U)$ for each RK stage. In general we observed better performance, especially in controlling overshoots, using the second variation, so we show all of our examples with the second variation.

A reference solution is computed using the standard fully explicit WENO scheme.

To effectively measure the gain of time by using the fractional step method, we use multiples of the explicit soundspeed-based timestep. So the timestep τ is computed via the CFL-condition of section 3.2:

$$\tau = c_o \cdot \frac{\max\limits_{i=0,..,m}(|\lambda_i|)}{h}$$

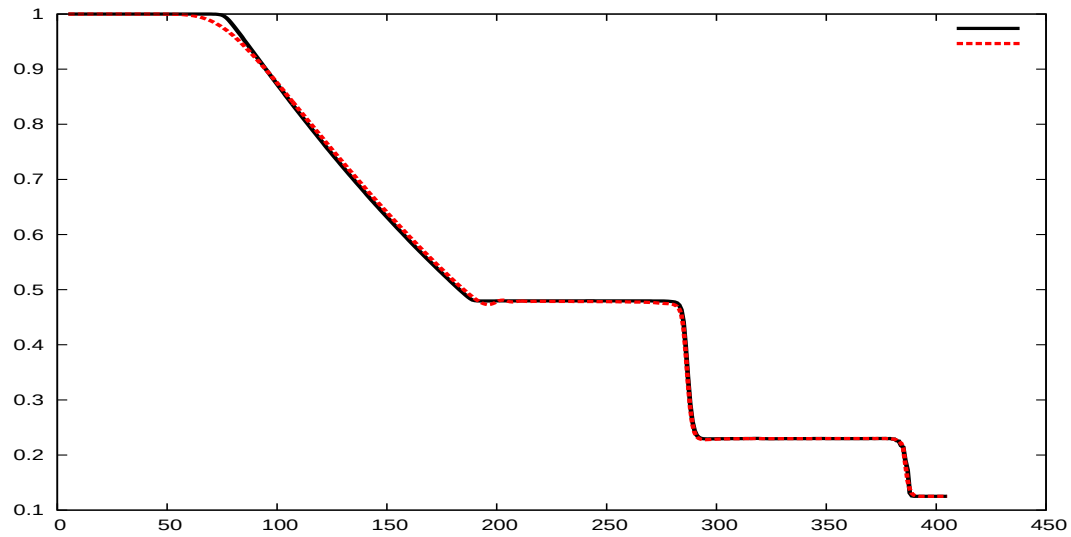
where c_o denotes the **Courant Number**.

4.4.1 SOD Shocktube 1D

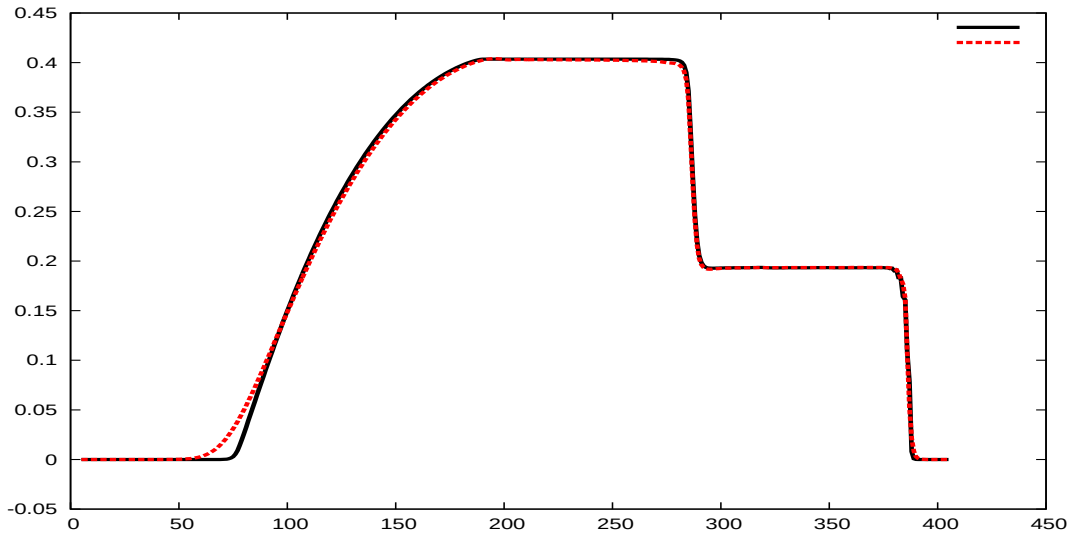
We use a computational domain of 200cm and 405 grid points. The initial conditions are

$$(\rho, u, P) = \begin{cases} (1, 0, 1) & \text{if } x \leq 50cm \\ (0.125, 0, 0.1) & \text{if } x \geq 50cm \end{cases}$$

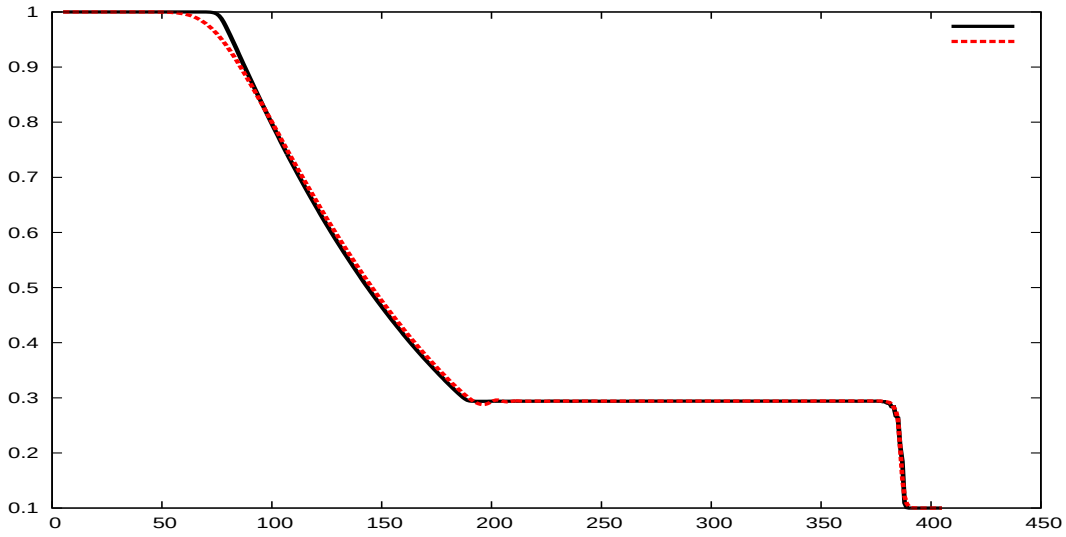
We compare the solution of the fractional step method (dotted red line) to our reference solution (blue line).
Since the proposed fractional step method is based on an implicit solution for the pressure, the temporal increment can be chosen greater than for the fully explicit WENO-scheme. We try various values for the Courant number c_o, observing that the explicit scheme becomes unstable at $c_o = 0.8$ ($\tau = 0.346s$) whereas our implicit scheme runs stable until $c_o = 1.0$ ($\tau = 0.433s$).



(a) Density

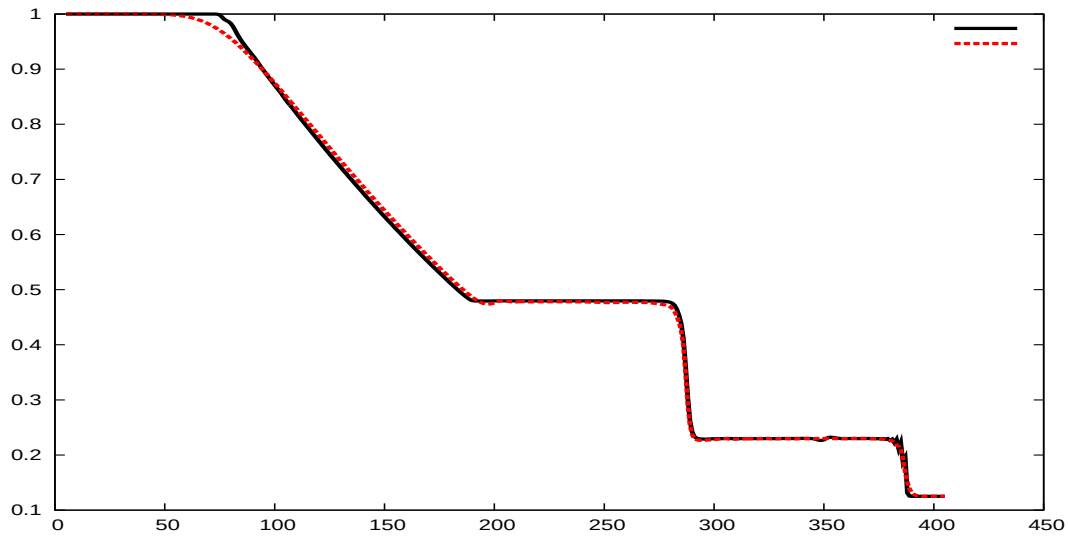


(b) Momentum

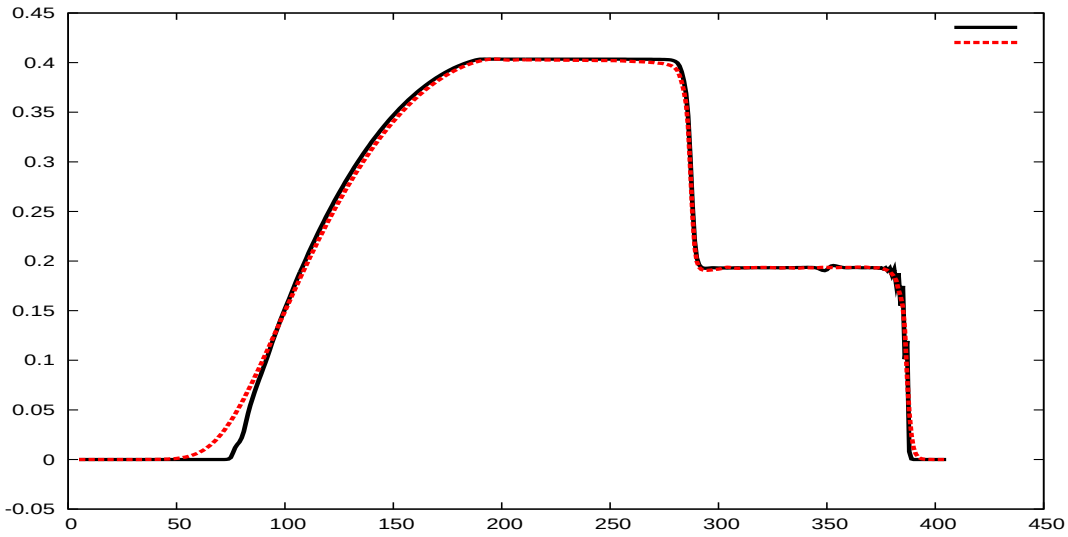


(c) Pressure

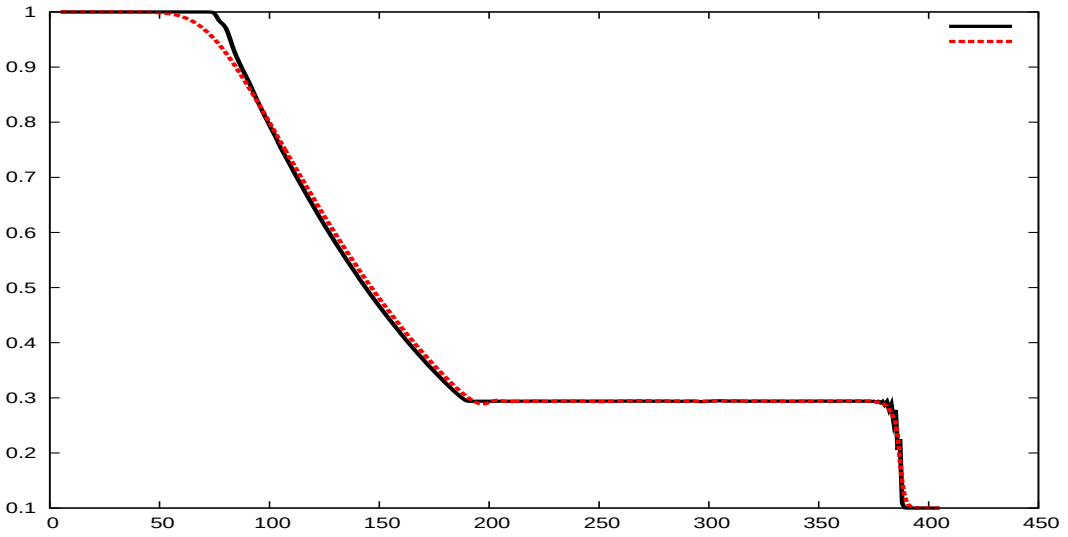
Figure 4.2: $c_o = 0.5$ $(\tau = 0.217s)$



(a) Density

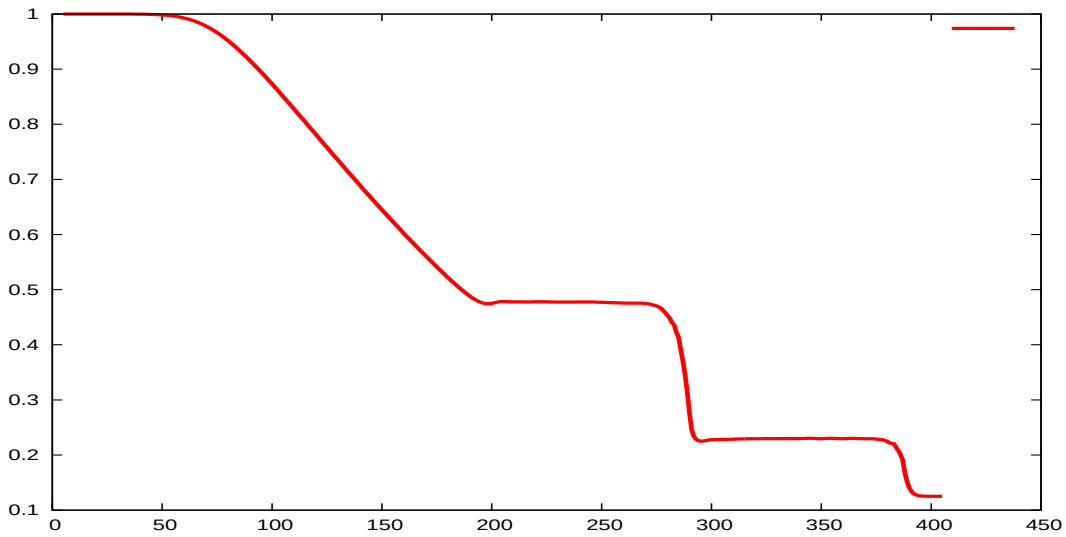


(b) Momentum

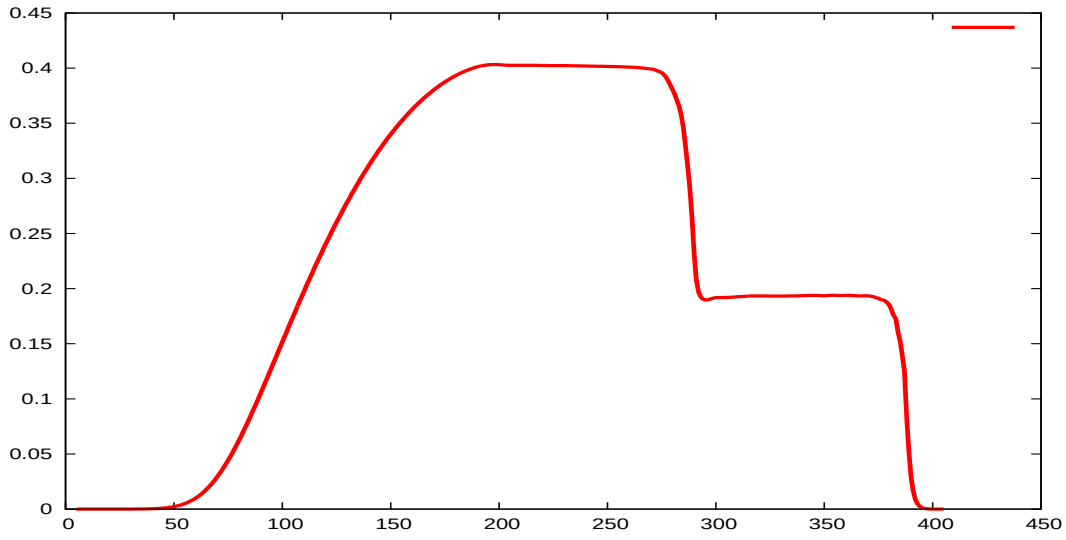


(c) Pressure

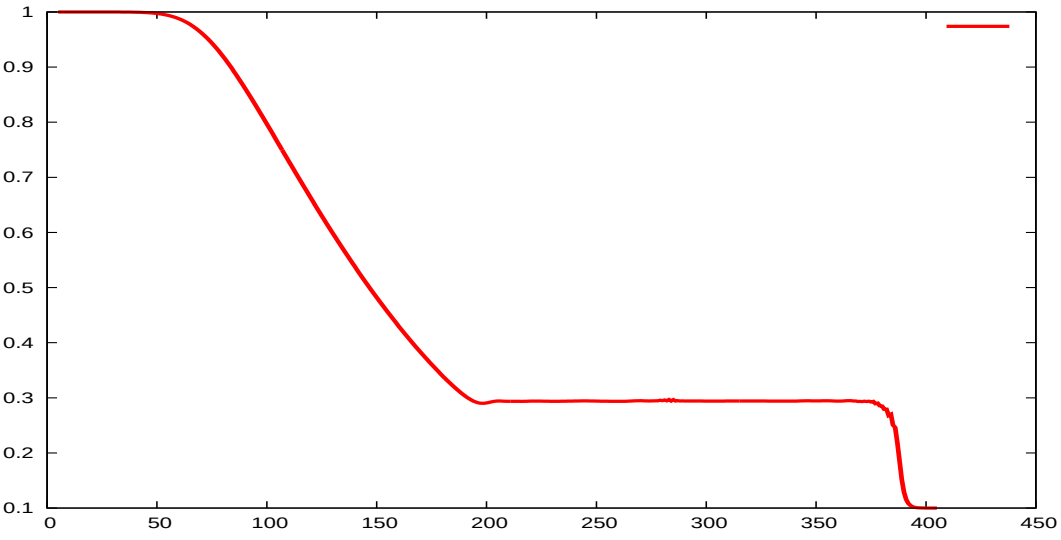
Figure 4.3: $c_o = 0.8$ ($\tau = 0.346s$): We see that the explicit solution slightly starts to oscillate.



(a) Density



(b) Momentum



(c) Pressure

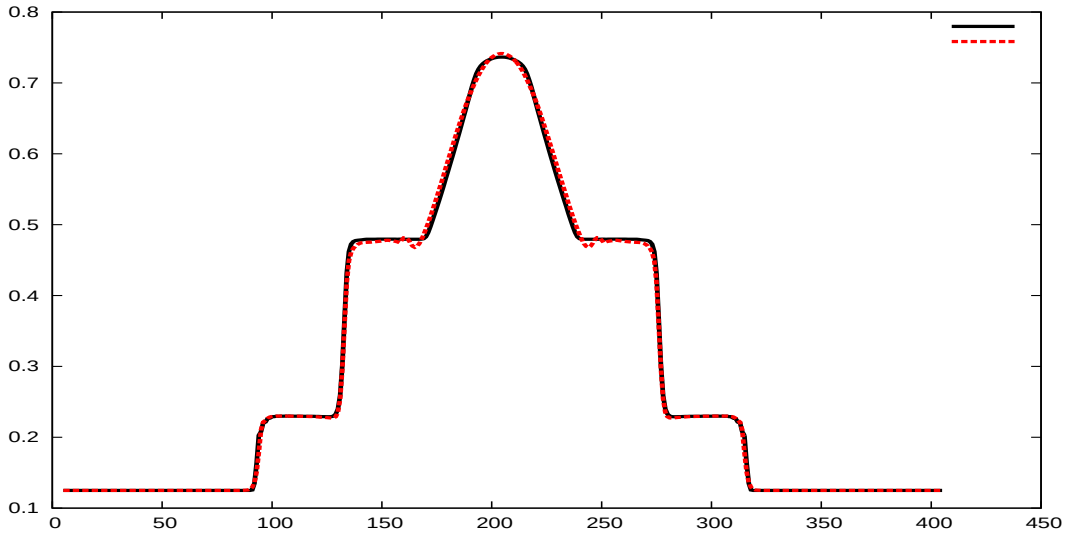
Figure 4.4: $c_o = 1.0$ ($\tau = 0.433s$): The fractional step method still yields stable results.

4.4.2 Double Shock Test

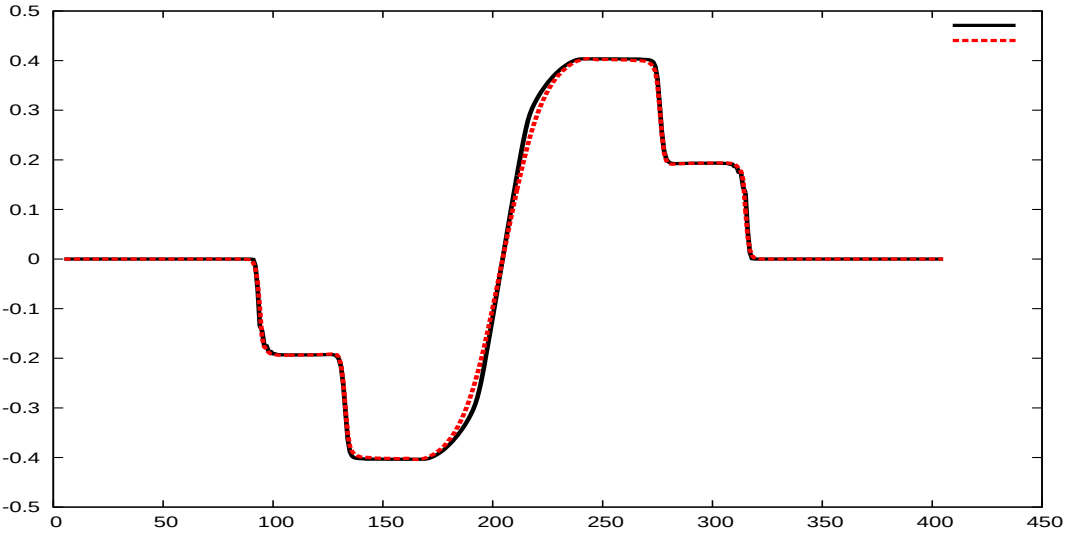
We use a computational domain of 200 cm and 405 grid points. The initial conditions create a double shock:

$$(\rho, u, P) = \begin{cases} (1, 0, 1) & \text{if } x \leq 30cm \\ (0.125, 0, 0.1) & \text{if } x \geq 30cm \text{ and } x \leq 70cm \\ (1, 0, 1) & \text{if } x \geq 70cm \end{cases}$$

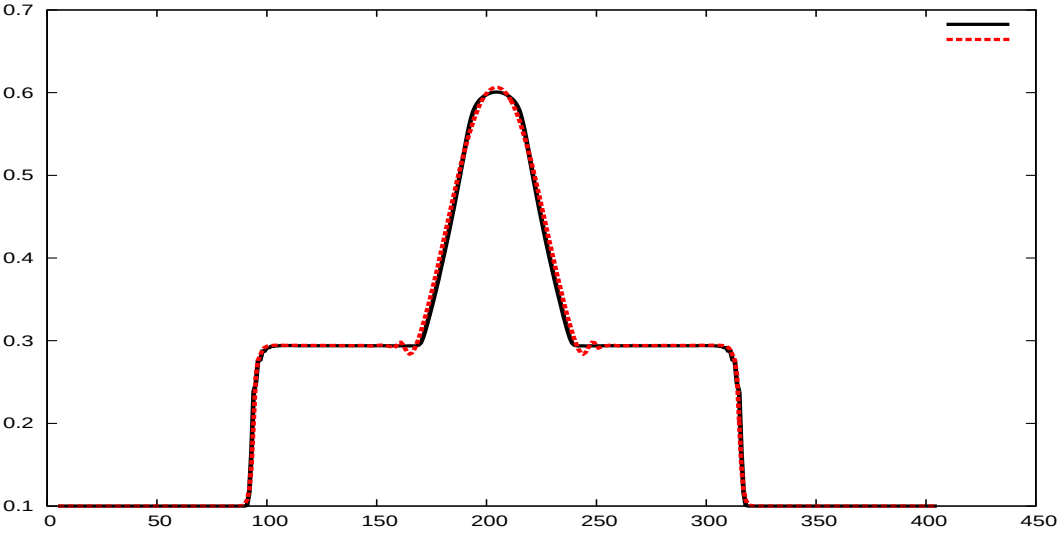
Again, we compare various values for c_o. The fractional step methods (dotted red line) yields stable results until $c_o = 1.0$ ($\tau = 0.433s$), whereas the standard explicit scheme (blue line) becomes unstable at $c_o = 0.9$ ($\tau = 0.389s$).



(a) Density



(b) Momentum



(c) Pressure

Figure 4.5: $c_o = 0.5$ $(\tau = 0.217s)$

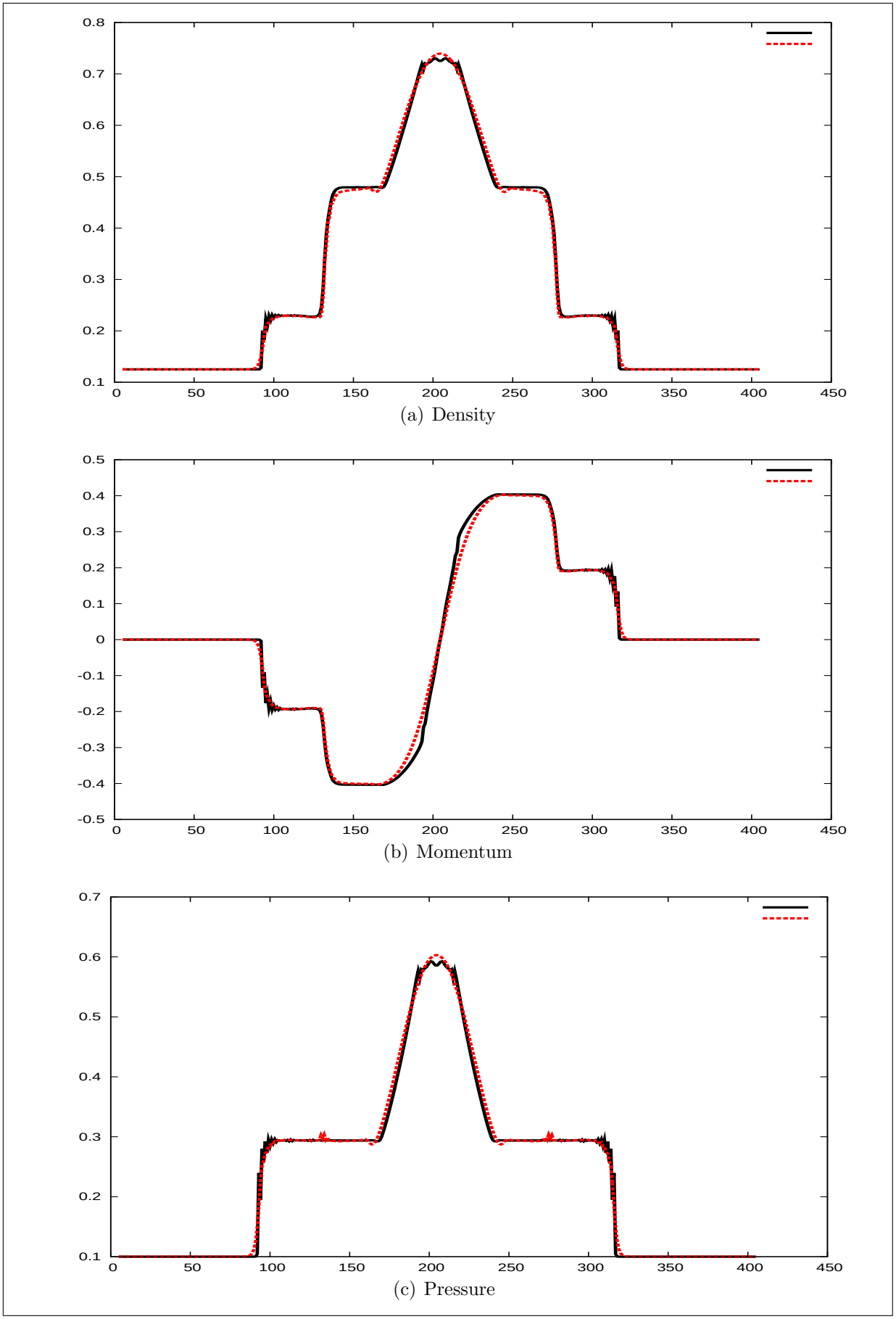
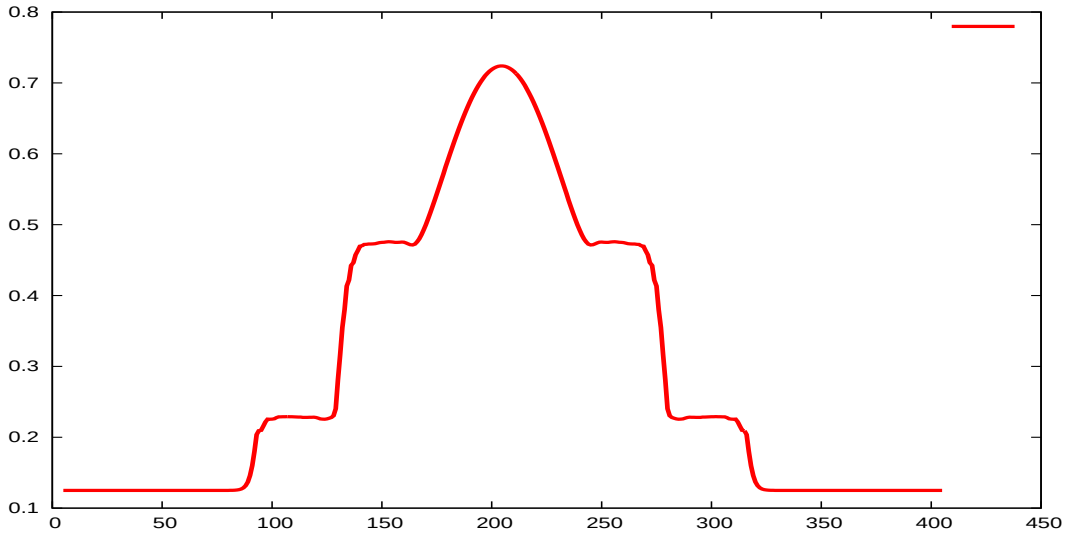
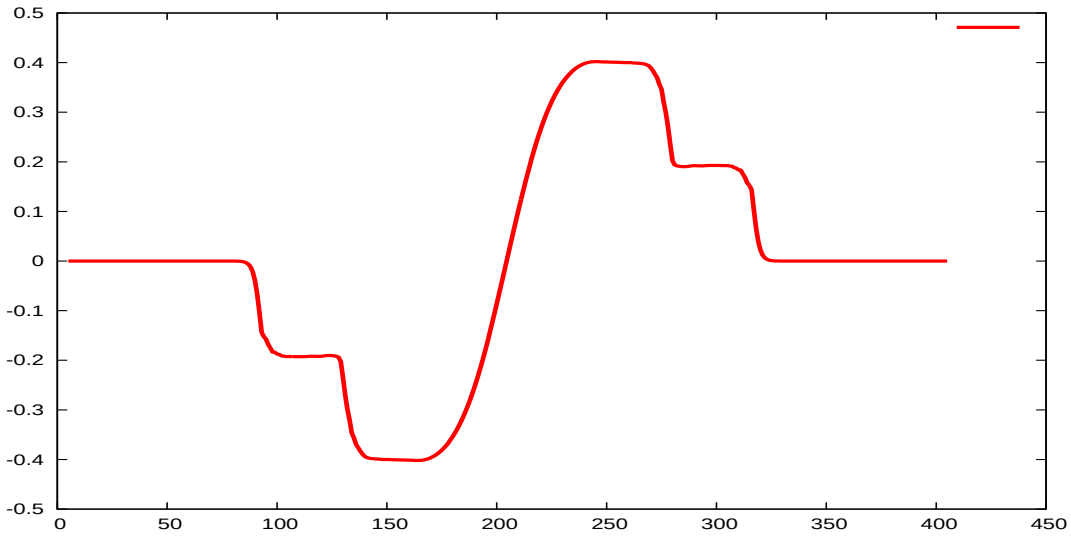


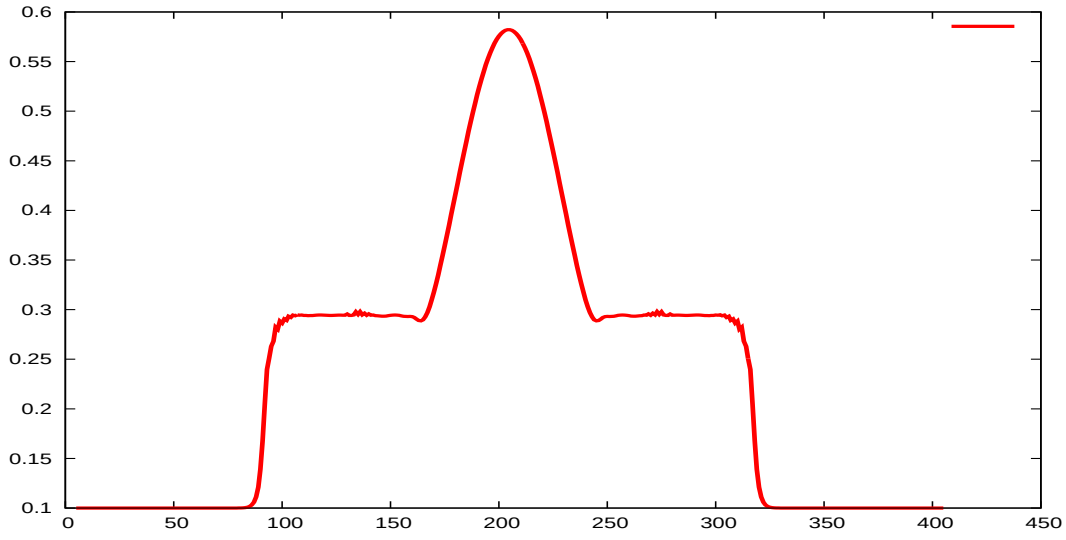
Figure 4.6: $c_o = 0.9$ ($\tau = 0.389s$): The explicit solution starts to oscillate.



(a) Density



(b) Momentum



(c) Pressure

Figure 4.7: $c_o = 1.0$ ($\tau = 0.433s$): The fractional step method still yields stable results.

4.4.3 Smooth Flow Test

To test the time step restriction of section 4.3, Kwatra et alii set up a smooth flow scenario dominated by acoustic waves (see [10]).
The initial conditions of this test are given by

$$u(x,0) = 0$$
$$P(x,0) = P_0 + \epsilon P_1(x)$$
$$P_1(x) = 6cos(2\pi x) + 10sin(4\pi x) \rho(x,0) = \left(\frac{P(x,0)}{\rho_0}\right)^{\frac{1}{\gamma}} \rho_0$$

γ being the adiabatic index of the gas and $\rho_0 = 1$, $P_0 = 10^3$, $\epsilon = 1$.
For better comparison, we clamp our time step to be a fixed multiple of the explicit time step based on the soundspeed based CFL-condition of section 3.2.

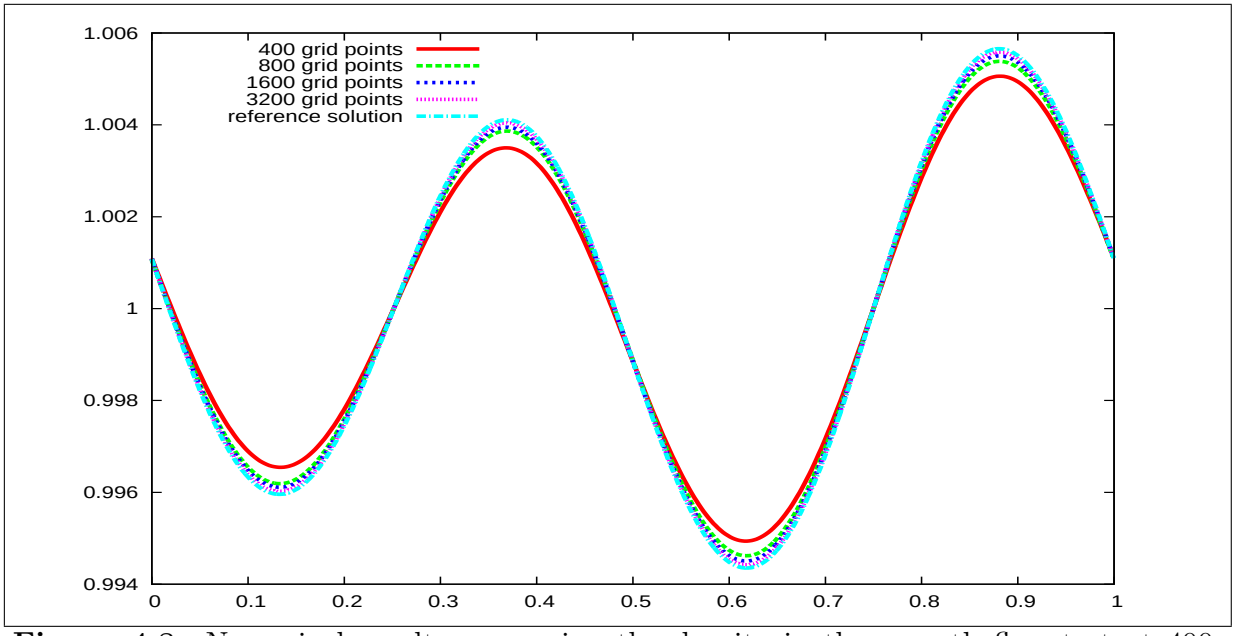


Figure 4.8: Numerical results comparing the density in the smooth flow test at 400, 800, 1600 and 3200 grid points with an effective sound speed based CFL number 3. The reference solution is computed using 3200 grid points and a CFL number of 0.5.

In Figure 4.8 we use 3 times the explicit time step and show convergence via grid resolution.

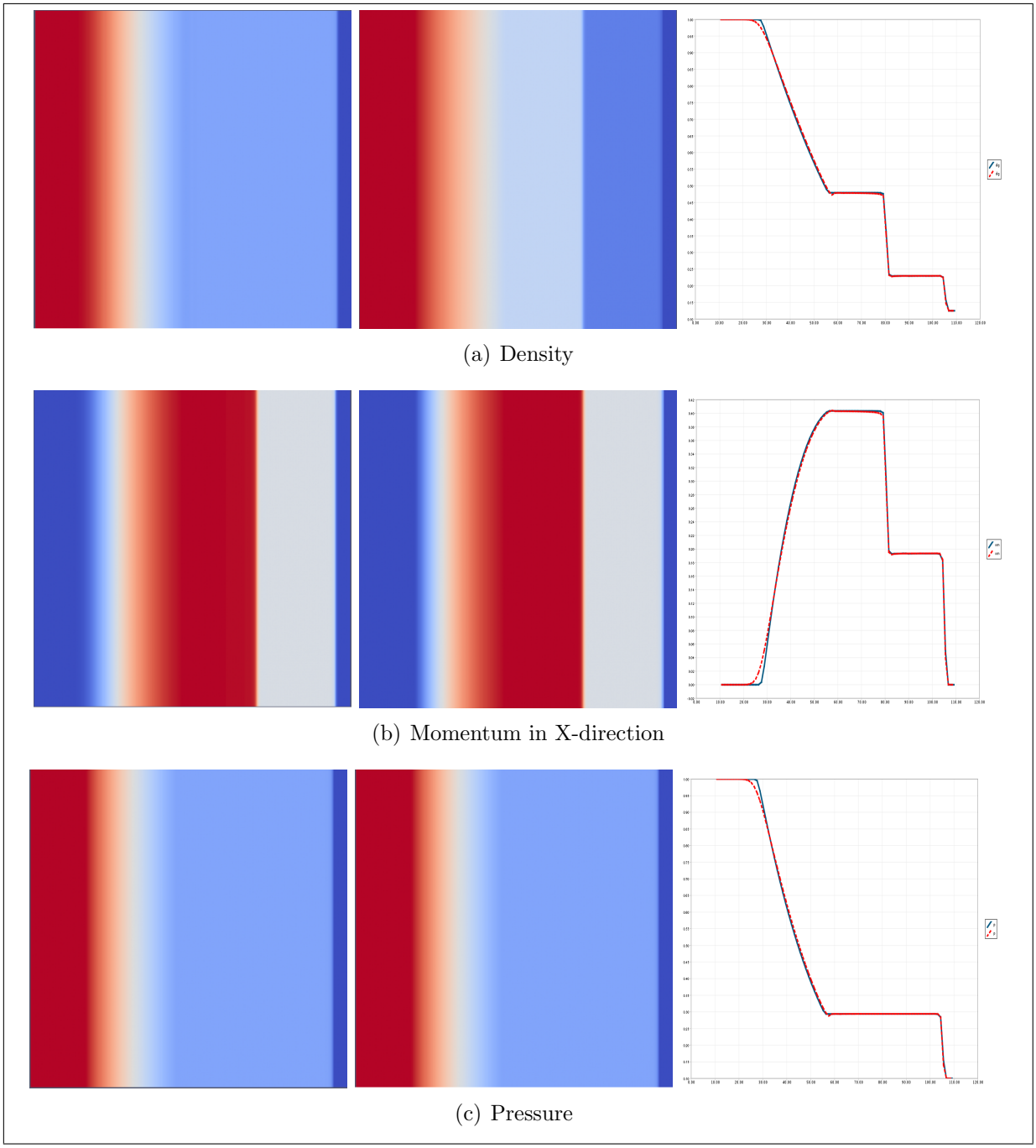
4.4.4 SOD Shocktube Test 2D

In order to validate our scheme in higher dimensions, we set up the Sod Shocktube test in 2D. We choose a computational domain of 100cm $\times$ 100cm and discretize it with 405

$\times$ 405 grid points. The initial conditions are analogous to those of the 1D Sod Shocktube test:

$$(\rho, u, v, P) = \begin{cases} (1, 0, 0, 1) & \text{if } x \leq 30cm \\ (0.125, 0, 0, 0.1) & \text{if } x \geq 30cm \text{ and } x \leq 70cm \\ (1, 0, 0, 1) & \text{if } x \geq 70cm \end{cases}$$

Below we see the numerical results of this test. The left figure shows the solution computed with the implicit scheme, next to it is the reference solution. On the right hand side we see a 1D slice, with the implicit solution shown in red. The comparison of various values for c_o yields results which are analogous to the 1D case.



(a) Density

(b) Momentum in X-direction

(c) Pressure

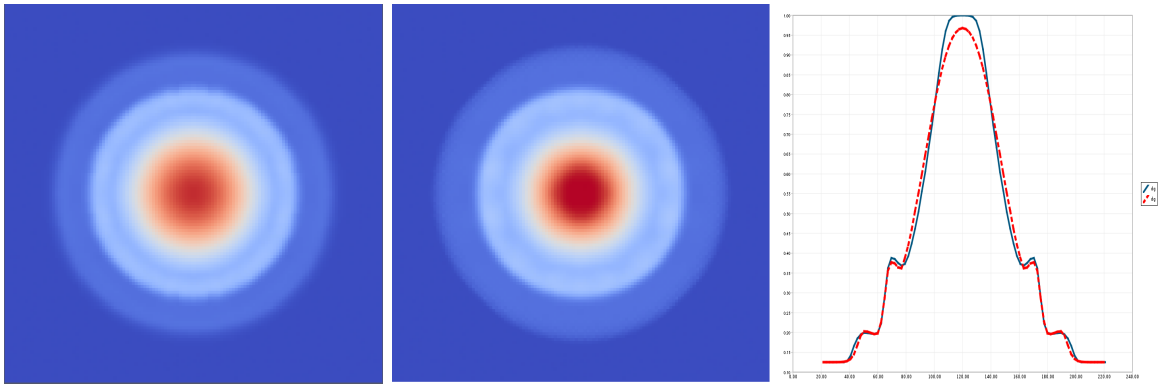
Figure 4.9: Sod Shocktube Test 2D

4.4.5 Circular Shock Test

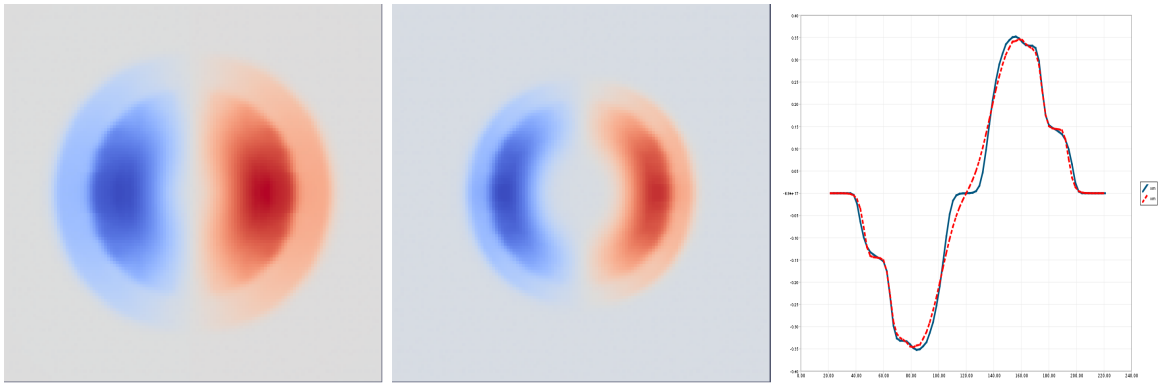
We use a computational domain of 200 cm × 200 cm and discretize it with 91 grid points in each direction. The circular shock test has an initial condition prescribed as

$$(\rho, u, v, P) = \begin{cases} (1, 0, 0, 1) & \text{if } r \leq 40cm \\ (0.125, 0, 0, 0.1) & \text{if } r \geq 40cm \end{cases}$$

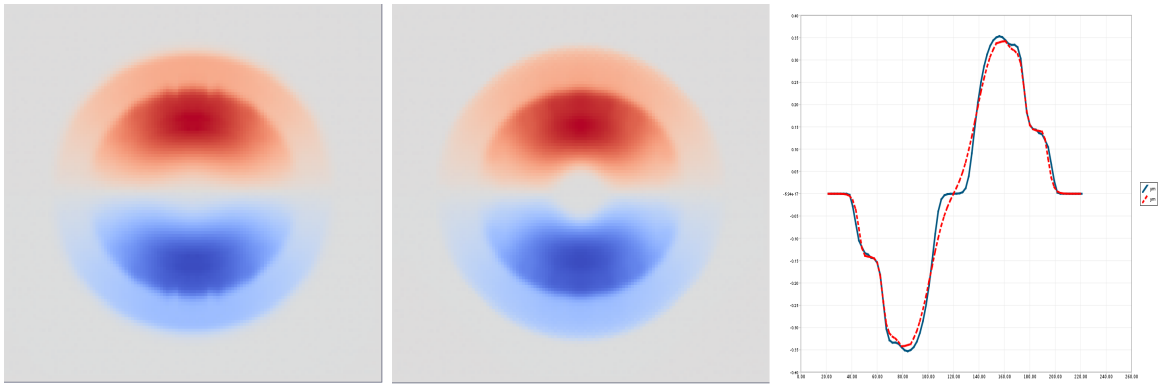
where $r = \sqrt{(100-x)^2 + (100-y)^2}$.
As before, we observe that the implicit method remains stable at higher values for c_o than the explicit method.
The plots below show the implicit solution on the left hand side, next to it the explicit solution and a 1D-slice with the reference solution plotted in red.



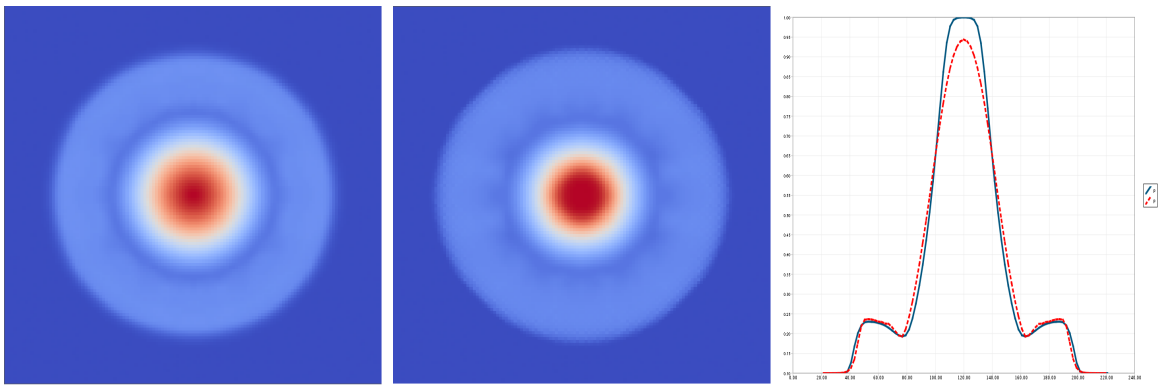
(a) Density



(b) Momentum in X-direction

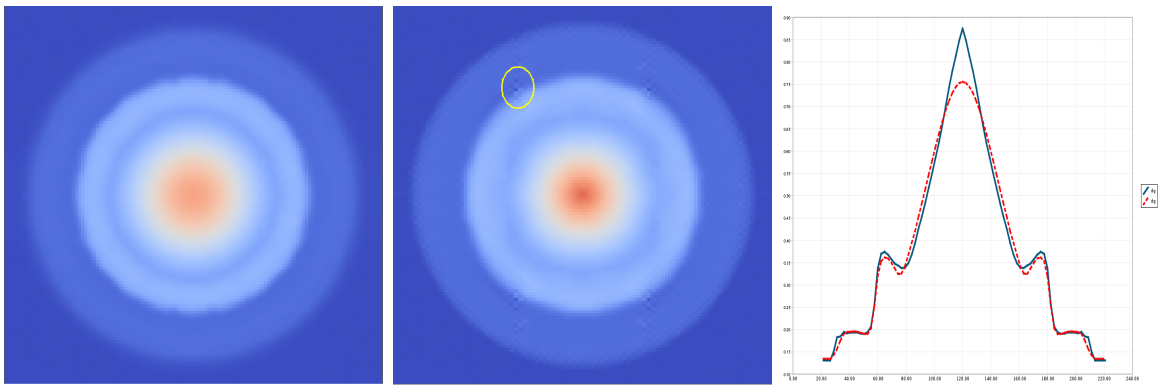


(c) Momentum in Y-direction

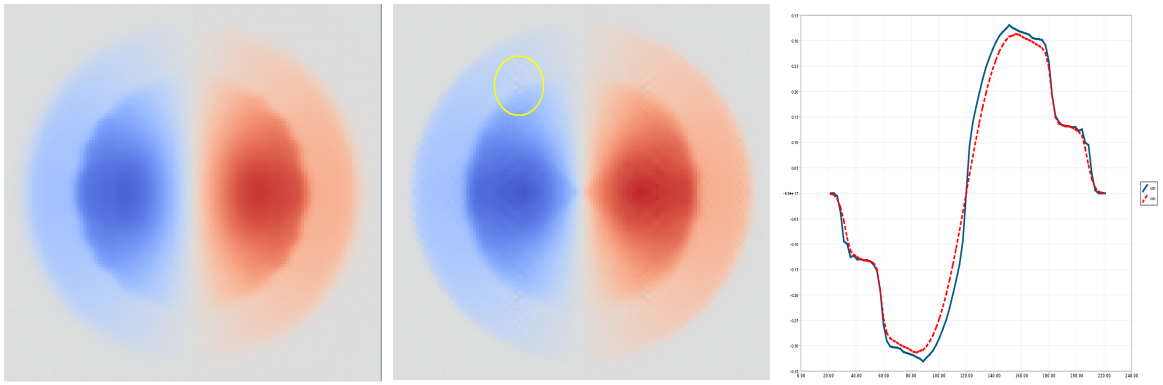


(d) Pressure

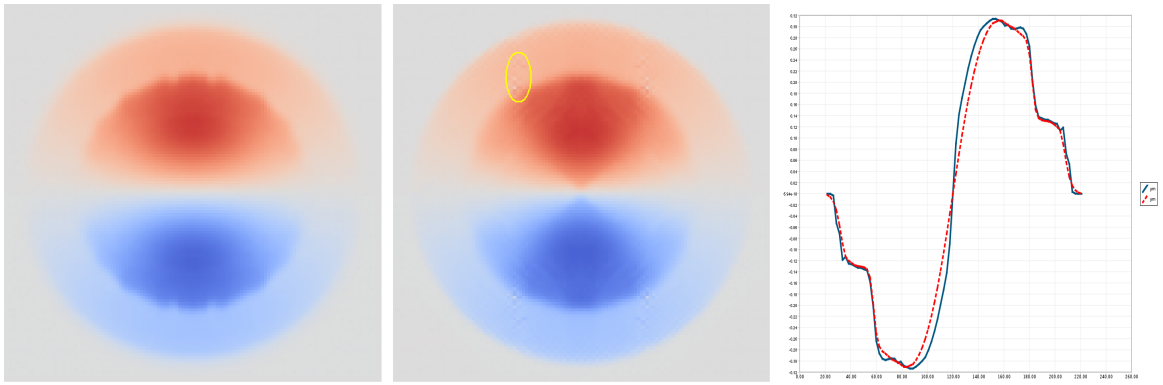
Figure 4.10: $c_o = 0.5$ $(\tau = 0.9622s)$



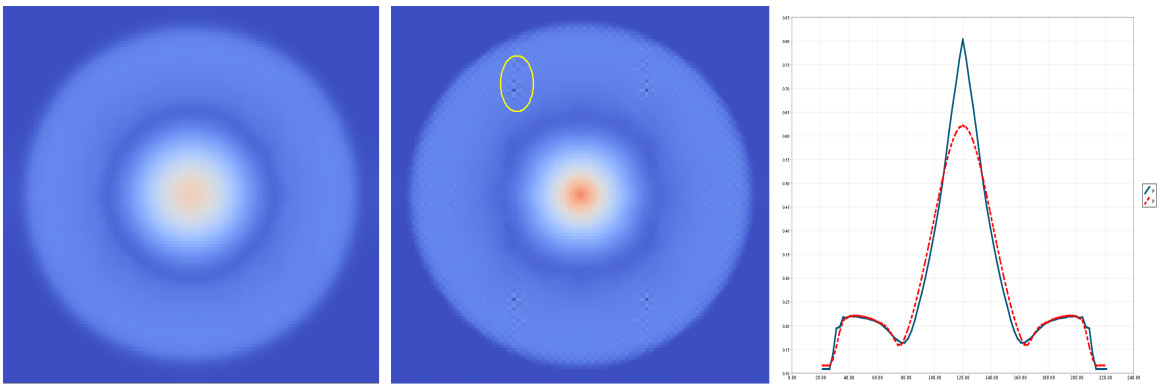
(a) Density



(b) Momentum in X-direction

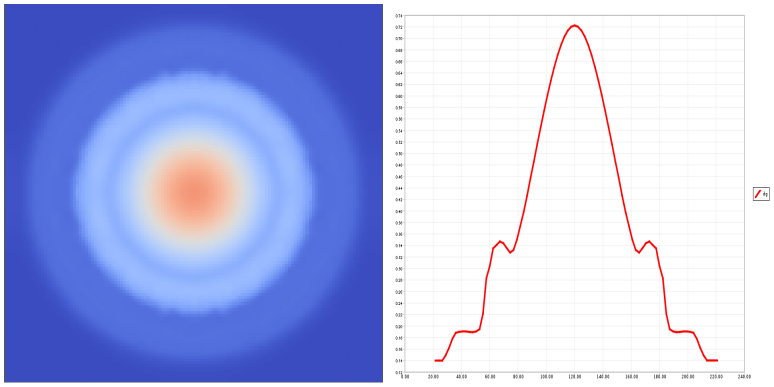


(c) Momentum in Y-direction

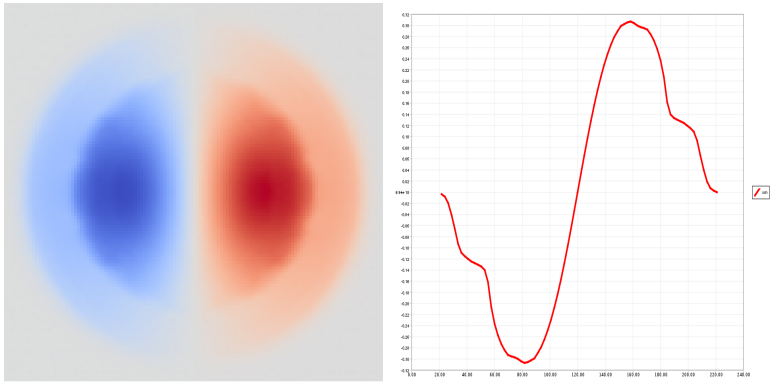


(d) Pressure

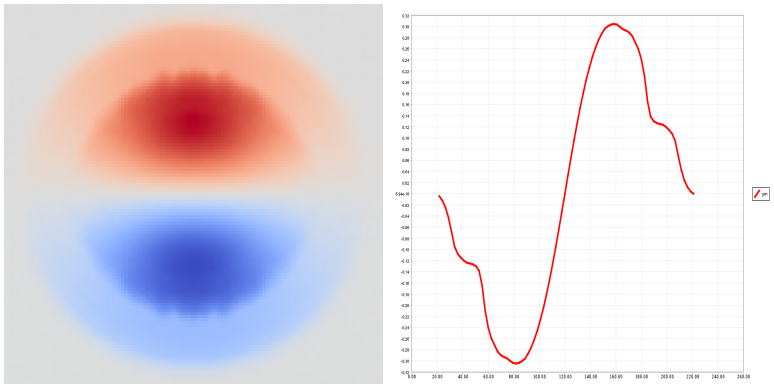
Figure 4.11: $c_o = 0.7$ ($\tau = 1.347s$). The explicit solution becomes unstable.



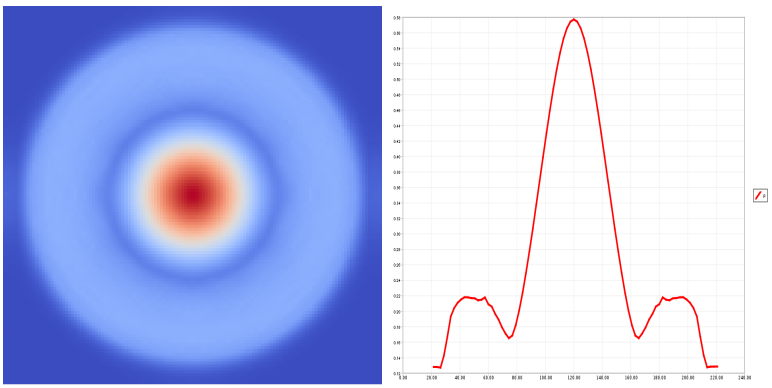
(a) Density



(b) Momentum in X-direction



(c) Momentum in Y-direction



(d) Pressure

Figure 4.12: $c_o = 1.0$ ($\tau = 1.924s$). The fractional step method still yields stable results.

4.4.6 Boundary conditions

In all tests, we impose the following conditions at boundaries in x-direction:

$$\begin{aligned} \nabla\rho &= 0 \\ \vec{u} &= 0 \\ \nabla(\rho e_{tot}) &= 0 \end{aligned}$$

In y-direction periodic boundary conditions are used.

The solution of the Poisson-like pressure evolution equation is calculated using Dirichlet boundary conditions. To increase the computational performance, P^n is used as an initial guess for the pressure at P^{n+1}.

To calculate $\nabla \cdot \vec{u}$ and ∇P, we interpolate $\vec{u}$ and P to the cell faces. Matching the no-slip boundary conditions in x-direction, we set

$$\vec{u}_{a-\frac{1}{2}} = -\vec{u}_{a+\frac{1}{2}} \tag{4.4.1}$$

$$\vec{u}_{b+\frac{1}{2}} = -\vec{u}_{b-\frac{1}{2}} \tag{4.4.2}$$

and P is extrapolated to $P_{a-\frac{1}{2}}$ and $P_{b+\frac{1}{2}}$.

Chapter 5

Extension to the Navier-Stokes Equations

Like Euler's Equations, the Navier-Stokes Equations also describe the motion of a fluid, but take more physical phenomena into account; viscosity, gravity and heat conduction are considered additionally to the pressure force.
The one-dimensional Navier-Stokes equations read

$$\begin{pmatrix} \rho \\ \rho u \\ \rho e_{tot} \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + P - \sigma \\ \rho e_{tot} u + P u - u \cdot \sigma + q \end{pmatrix}_x - \begin{pmatrix} 0 \\ \rho g \\ \rho g \end{pmatrix} = 0$$

σ denotes the viscous stress tensor and q denotes the thermal flux density.

We split the fluxes into an advective, a non-advective and a viscous part:

$$F_1(U) = \begin{pmatrix} \rho u \\ \rho u^2 \\ \rho e_{tot} u \end{pmatrix} \tag{5.0.1}$$

$$F_2(U) = \begin{pmatrix} 0 \\ P \\ P u \end{pmatrix} \tag{5.0.2}$$

$$F_3(U) = \begin{pmatrix} 0 \\ \sigma \\ \sigma u + q \end{pmatrix} \tag{5.0.3}$$

To obtain intermediate values $\rho^*, (\rho u)^*$ and $(\rho e_{tot})^*$ we update $F_1(U)$, $F_3(U)$ and add the gravity term:

$$\frac{\rho^* - \rho u^n}{\tau} = -\nabla \cdot (\rho u) \tag{5.0.4}$$

$$\frac{(\rho u)^* - (\rho u)^n}{\tau} = -\nabla \cdot (\rho u^2) + \nabla \cdot \sigma + \nabla \cdot q + \rho g \tag{5.0.5}$$

$$\frac{(\rho e_{tot})^* - (\rho e_{tot})^n}{\tau} = -\nabla \cdot (\rho e_{tot} u) + \nabla \cdot (u \cdot \sigma) + \nabla \cdot q + \rho g \tag{5.0.6}$$

Since neither the pressure nor the viscosity tensor affect the continuity equation, we have $\rho^* = \rho^{n+1}$.

All the fluxes are additive, so we can treat them separately. The advective flux $F_1(U)$ is updated via the Complementary Projection method, whereas the viscous part of $F_3(U)$ is calculated by centrally differenced interpolation:

$$(\sigma_x)_i = \frac{\sigma_{i+\frac{1}{2}} - \sigma_{i-\frac{1}{2}}}{h_x}$$
$$((u\sigma)_x)_i = \frac{(u\sigma)_{i+\frac{1}{2}} - (u\sigma)_{i-\frac{1}{2}}}{h_x}$$

where

$$(\sigma_x)_{i-\frac{1}{2}} = -\frac{1}{16}\sigma_{i-2} + \frac{9}{16}\sigma_{i-1} + \frac{9}{16}\sigma_i - \frac{1}{16}\sigma_{i+1}$$
$$((u\sigma)_x)_{i-\frac{1}{2}} = -\frac{1}{16}(u\sigma)_{i-2} + \frac{9}{16}(u\sigma)_{i-1} + \frac{9}{16}(u\sigma)_i - \frac{1}{16}(u\sigma)_{i+1}$$

Analogously we calculate $\nabla \cdot q$.

Once we obtained the intermediate values, we apply the pressure correction of section 4:

$$\frac{(\rho\vec{u})^{n+1} - (\rho\vec{u})^*}{\tau} = -\nabla P \tag{5.0.7}$$

$$\frac{(\rho e_{tot})^{n+1} - (\rho e_{tot})^*}{\tau} = -\nabla \cdot (P\vec{u})^{n+1} \tag{5.0.8}$$

Remark 5.0.1. For reasons of physics, we use **Neumann boundary conditions** for our Poisson-like equation:

$$\nabla P = 0 \tag{5.0.9}$$

5.1 Timestep Restriction

N.Kwatra et alii suggest in [10] equation (4.3.7) to compute an optimal timestep. Since it is derived for Euler's equations, it does not include restrictions due to viscosity or thermal diffusivity.

Furthermore, the derivation follows the lines of a timestep restriction used in Navier-Stokes solvers for incompressible flows. Since our flows are compressible, this timestep control is not applicable to our simulations.

The governing timestep restrictions in our simulated processes are imposed by the thermal diffusivity, viscosity and the velocity of the fluid:

$$\tau_{th}\frac{\kappa}{h_x^2 c_p \rho} \leq 1 \tag{5.1.1}$$

$$\tau_v\frac{\nu}{h_x^2} \leq 1 \tag{5.1.2}$$

$$\tau_f\frac{u}{h_x} \leq 1 \tag{5.1.3}$$

c_p denotes the specific heat at constant pressure, κ the corresponding diffusion coefficient and ν the kinematic viscosity. See [2].

In two dimensions, this becomes

$$\tau_{th}\frac{\kappa}{min(h_x,h_y)^2 c_p \rho} \leq 1 \tag{5.1.4}$$

$$\tau_v\frac{\nu}{(min(h_x,h_y)^2} \leq 1 \tag{5.1.5}$$

$$\tau_f\frac{\sqrt{u^2+v^2}}{min(h_x,h_y)} \leq 1 \tag{5.1.6}$$

We fix our timestep τ to be the minimum of those above, and to ensure stability we follow the lead of [6] and use $\tau_f^{'} = \sqrt{At}\cdot\tau_f$, where At denotes the Atwood number. For our purposes, $\sqrt{At} \approx 0.5$:

$$\tau = min(\tau_{th}, \tau_v, \tau_f^{'}) \tag{5.1.7}$$

5.2 Application to Astrophysics: Convection in Stars

5.2.1 Convective Instability - The Schwarzschild Criterion

Imagine a bubble in a fluid. The density inside the bubble is given by

$$\rho' = \rho + \left(\frac{\partial \rho}{\partial r}\right)_{ad} \Delta r$$

where r denotes the radius and the subscript says that the process is **adiabatic**, i.e. there is no heat exchange with the surroundings.

The density in the bubble's exterior is

$$\overline{\rho} = \rho + \frac{\partial \rho}{\partial r} \Delta r$$

Naturally, the bubble rises (and therefore convection starts) if the density inside the bubble is lower than it is in its surroundings:

$$\rho' \leq \overline{\rho} \Rightarrow \left(\frac{\partial \rho}{\partial r}\right)_{ad} < \frac{\partial \rho}{\partial r}$$

Thermodynamical relations (see for example [2]) lead to

$$\left(\frac{\partial T}{\partial r}\right)_{ad} > \frac{\partial T}{\partial r} \tag{5.2.1}$$

We define the real (non-adiabatic) temperature gradient as

$$\nabla := \frac{\partial ln(T)}{\partial ln(P)}$$

and the adiabatic temperature gradient

$$\nabla_{ad} := \left(\frac{\partial ln(T)}{\partial ln(P)}\right)_{ad}$$

Having a closer look at equation 5.2.1 we deduce

$$\frac{\partial T}{\partial r} = \frac{\partial T}{\partial P}\frac{\partial P}{\partial r} = T\frac{\partial ln(T)}{\partial ln(P)}\frac{\partial ln(P)}{\partial r} = T\frac{\partial ln(P)}{\partial r}\nabla$$

and

$$\left(\frac{\partial T}{\partial r}\right)_{ad} = \left(\frac{\partial T}{\partial P}\right)_{ad}\frac{\partial P}{\partial r} = T\left(\frac{\partial ln(T)}{\partial ln(P)}\right)_{ad}\frac{\partial ln(P)}{\partial r} = T\frac{\partial ln(P)}{\partial r}\nabla_{ad}$$

Inserting this into 5.2.1 and dividing by $-\frac{\partial ln(P)}{\partial r}$ leads to

$$\nabla > \nabla_{ad} \tag{5.2.2}$$

This is the **Schwarzschild criterion** for convective instability.

5.2.2 Physical setting

According to [15], we specify a hydrostatic configuration, which is unstable against convection.
From the outset we assume the gas to be ideal γ-law gas with $\gamma = \frac{5}{3}$. The volume expansion coefficient is given by $\alpha = \frac{1}{1-\gamma}$ and the specific heat at constant pressure is $c_P = R_{gas}(1+\alpha)$.
These data suffice to determine the adiabatic temperature gradient

$$\left(\frac{\partial T}{\partial x}\right)_{ad} = \frac{g}{c_P} \tag{5.2.3}$$

once we have specified the downward pointing constant gravity g; x is the depth coordinate.
We fix the temperature at the top and define a function $b(x)$ to be the ratio of the actual temperature gradient to the adiabatic one:

$$\frac{\partial T}{\partial x} = b(x)\left(\frac{\partial T}{\partial x}\right)_{ad} \tag{5.2.4}$$

Obviously, if we choose $b(x) < 1$, our model is unstable to convection.
We can integrate equation 5.2.4 straightforwardly and the run of temperature - and therefore of energy - is specified.
We specify the density at the top and eliminate the pressure p in the equation of hydrostatic equilibrium. So we get an equation for ρ which can be integrated. We obtain P from the equation of state.
We start out at a velocity initially zero and apply a sinusoidal initial perturbation.
To complete our model we need the two viscosity coefficients. See [15] on how this is done.

5.2.3 Simulations

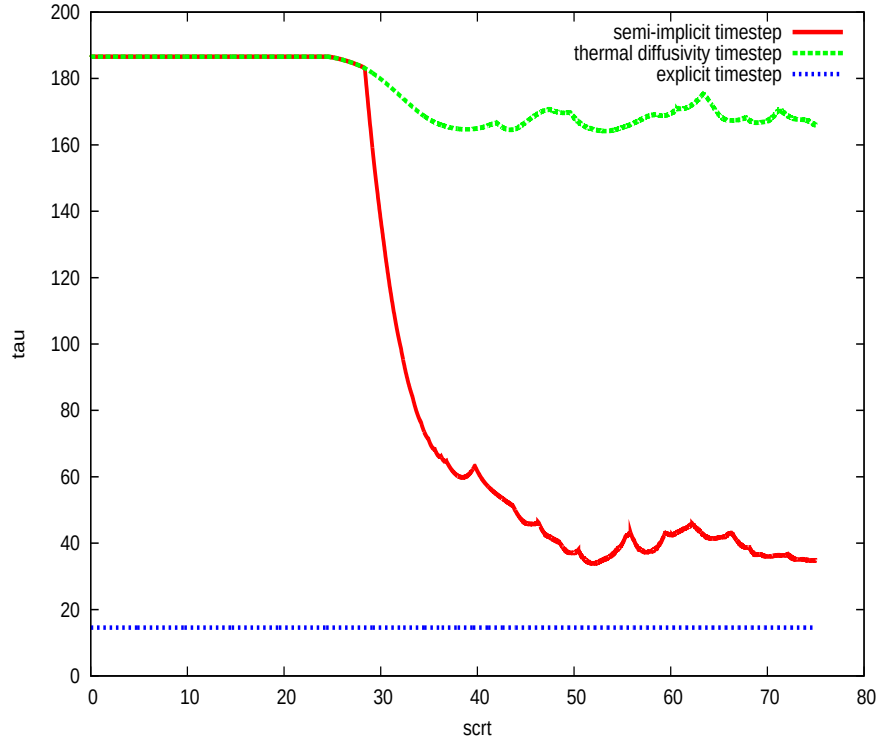
The initial set-up was implemented by Zaussinger, F. [1].
Simulations are run with 100x101 and 200x201 grid points. The height of the box, and therefore the spatial increment is determined by the physical setting - our box covers one

pressure scale height. This is the vertical distance over which the pressure changes by a factor of e.
The function $b(x)$ is specified as

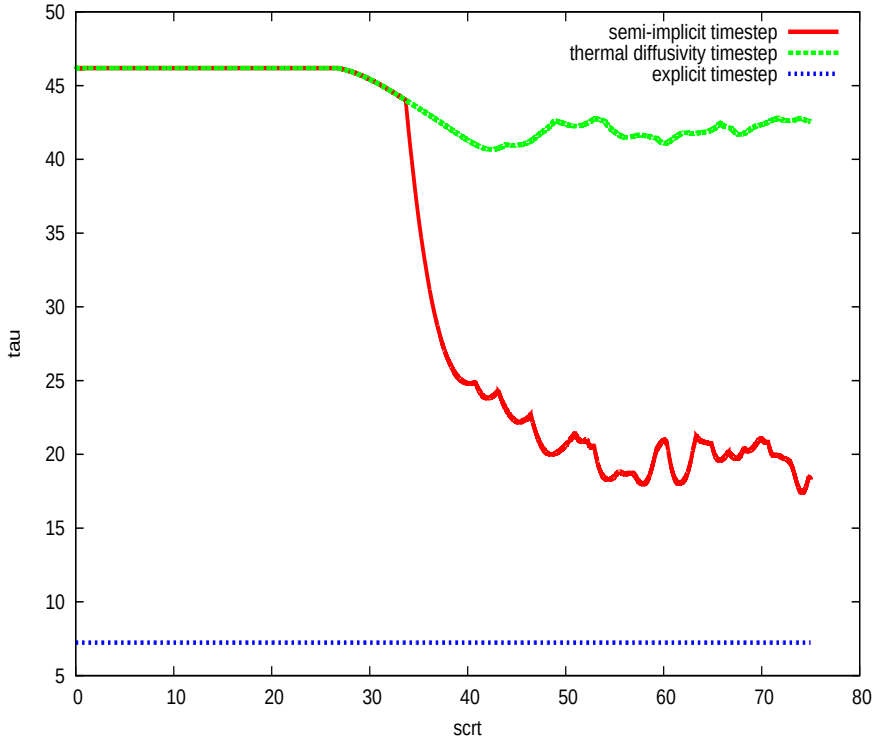
$$b(x) \equiv 2.0$$

We compare the timesteps of the fractional step method to the explicit timestep. We see that the timestep of the low resolution simulation starts out at much higher values than the simulation with 200x201 grid points. This is due to the fact that the spatial increment enters quadratically when computing τ_{th} and τ_v.

On the abscissa the time is specified in units of the **sound crossing time (scrt)**. This is the medium time needed for a sound wave to travel through the domain.



(a) 100 x 101 grid points



(b) 200 x 201 grid points

Figure 5.1: This plot shows evolution of the timesteps in the simulations.We see that at first the thermal diffusivity restricts τ until the velocities are high enough to pose a greater restriction. Due to $min(h_x, h_y)^2$ in equation (5.1.4), the initial time step of the 200x201 points simulation is just a fourth of the τ of the simulation with lower resolution. The blue line denotes the explicit timestep.

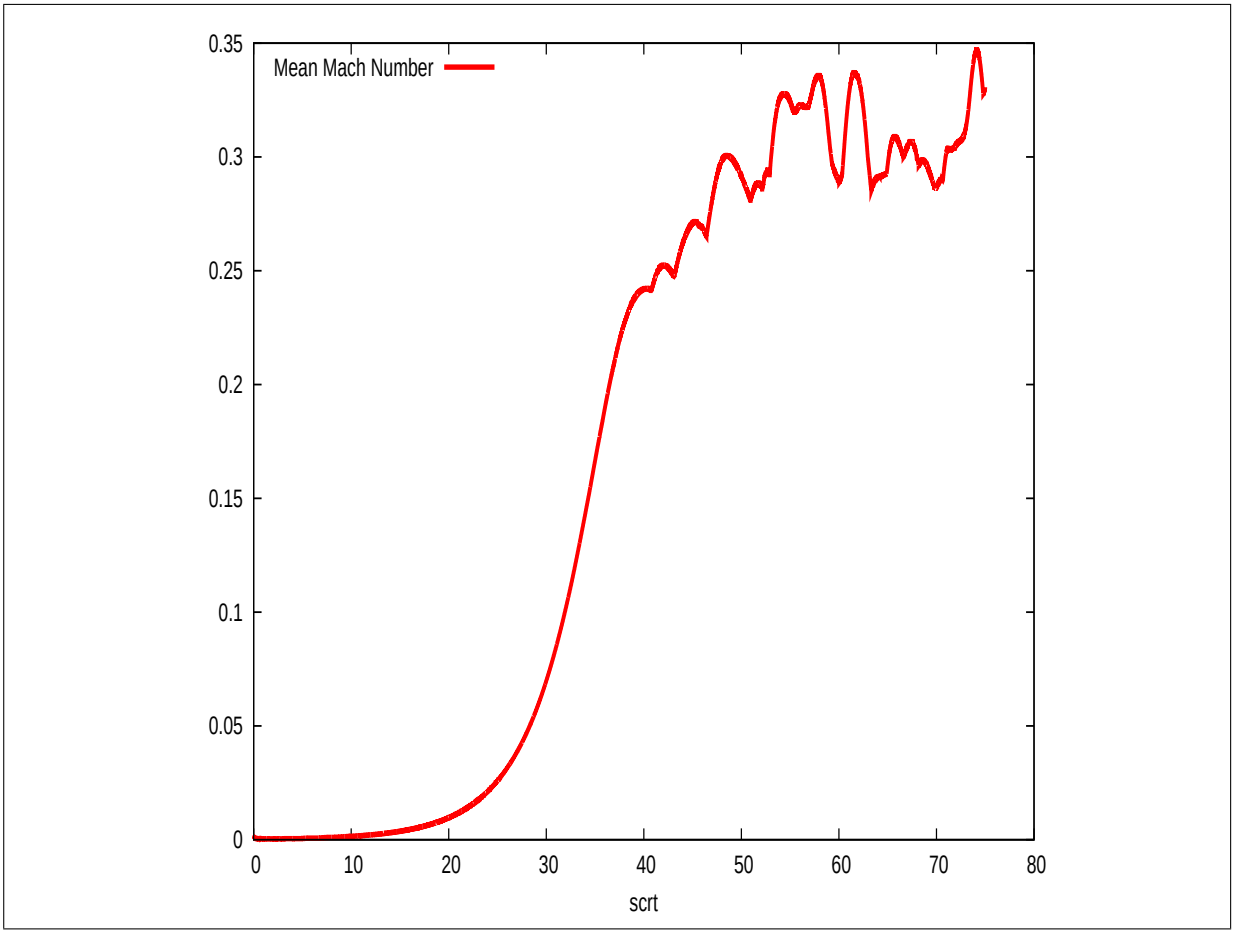
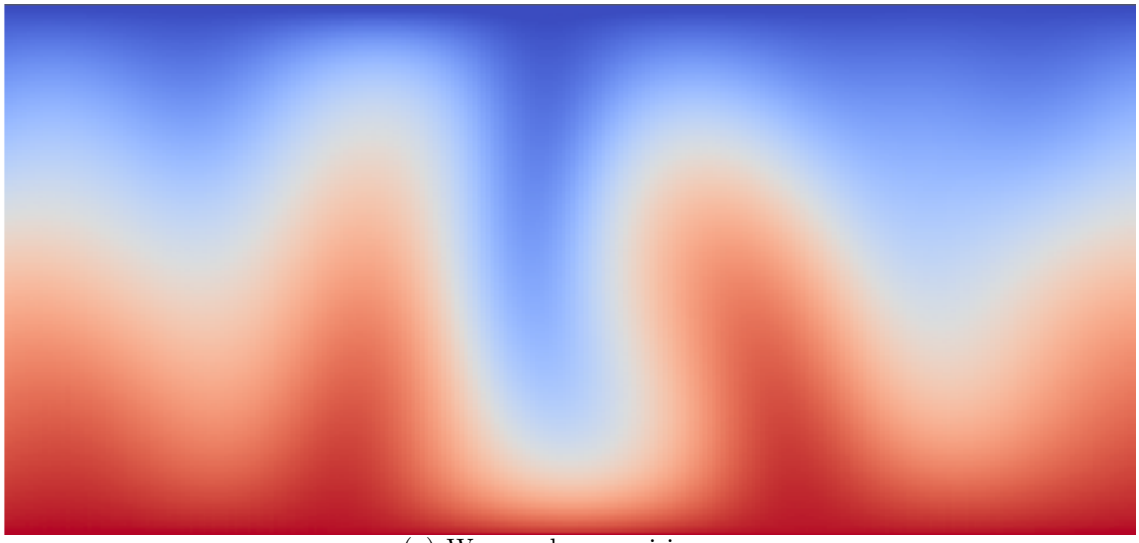
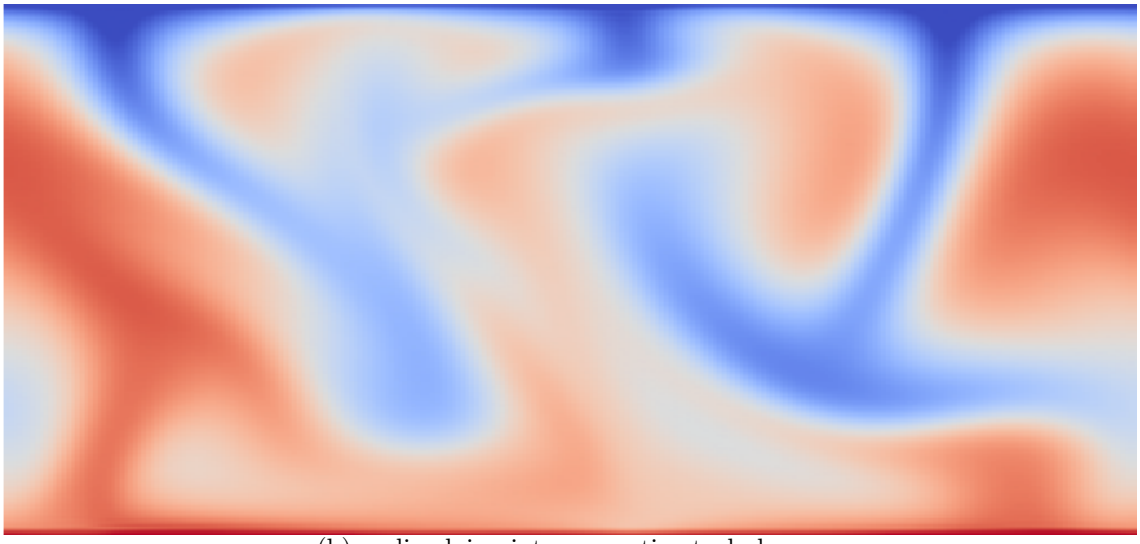


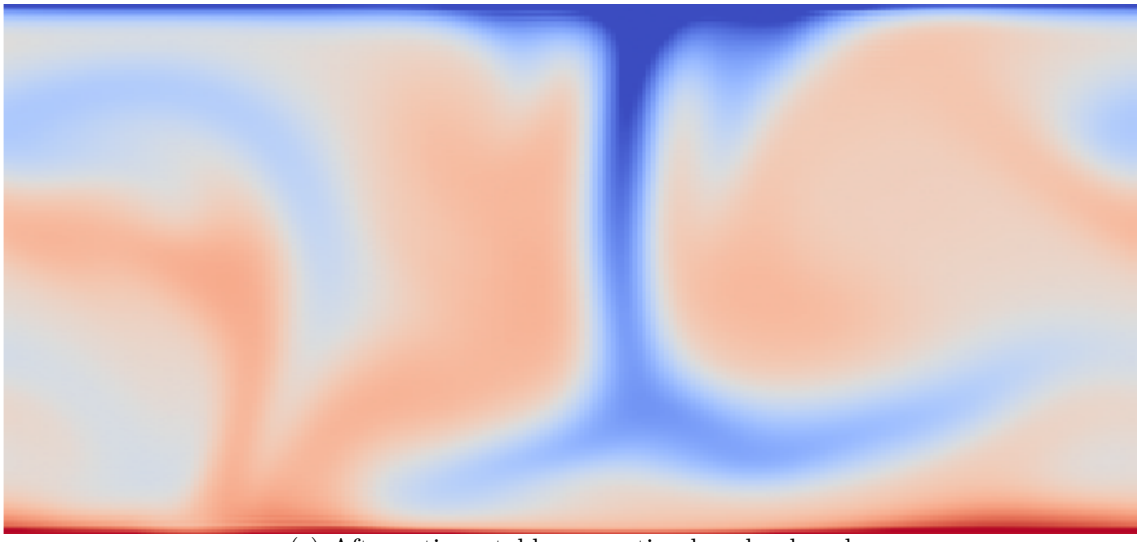
Figure 5.2: This plot shows the Mach Number of the simulations above. Comparing it to the timesteps, we see that especially in the Low Mach region we use significantly higher timesteps than permitted in the explicit scheme.



(a) We see plumes arising ...



(b) .. dissolving into convective turbulences.



(c) After a time stable convection has developed.

Figure 5.3: Snapshots of the simulation with 200x201 grid points. Plotted is the potential temperature t_{pot}.

5.3 Application to Astrophysics: Semiconvection in Stars

In massive stars, there exists a **semiconvection zone**, a chemically inhomogeneous zone between the convective core and the radiative envelope. Though semiconvection is an important mechanism for mixing chemical elements in late stages of stellar evolution, there have been few numerical simulations of semiconvective scenarios.
Since recent observations of the supernova SN1987A have been at odds with models of late stages of stellar evolution and one main suspect are the theories of elemental mixing used in those models, the interest in modelling semiconvection has grown.

The following deduction is taken from [1].

We consider a system with two fluids, with partial density ρc and density $(1-c)\rho$ respectively, where c denotes the **relative concentration**.
To describe the evolution of c, we use an advection-diffusion concentration equation,

$$\frac{\partial c}{\partial t} = -\vec{u} \cdot \nabla c + \kappa_c \nabla^2 c \tag{5.3.1}$$

,κ_c denoting the diffusion coefficient, whereas for ρ the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{5.3.2}$$

holds.
We need to derive an equation for the partial density ρc:

$$\frac{\partial (\rho c)}{\partial t} = \rho \frac{\partial c}{\partial t} + c \frac{\partial \rho}{\partial t}$$

Using equations (5.3.1) and (5.3.2) we get

$$\frac{\partial (\rho c)}{\partial t} = \rho \cdot \left(-\vec{u} \cdot \nabla c + \kappa_c \nabla^2 c\right) + c \cdot (-\nabla \cdot (\rho \vec{u}))$$

Basic vector calculus states that

$$\rho \vec{u} \cdot \nabla c + c \cdot \nabla (\rho \vec{u}) = \nabla \cdot (\rho \vec{u} \cdot c)$$

Thus we arrive at

$$\frac{\partial (\rho c)}{\partial t} = -\nabla \cdot (\rho \vec{u} \cdot c) + \rho \kappa_c \nabla^2 c \tag{5.3.3}$$

We want to model a mixture consisting of two chemical species with relative concentrations

c_1 and $c_2 = 1 - c_1$ respectively. The following equations describe the evolution of mass:

$$\frac{\partial \rho c_1}{\partial t} = -\nabla \cdot (\rho \vec{u} \cdot c_1) + \rho \kappa_{c_1} \nabla^2 c_1 \tag{5.3.4}$$

$$\frac{\partial \rho c_2}{\partial t} = -\nabla \cdot (\rho \vec{u} \cdot c_2) + \rho \kappa_{c_2} \nabla^2 c_2 \tag{5.3.5}$$

Obviously,

$$c_1 + c_2 = 1 \tag{5.3.6}$$

and

$$\nabla c_1 + \nabla c_2 = 0 \tag{5.3.7}$$

hold.
It can be shown that

$$\kappa_{c_1} = \kappa_{c_2} \tag{5.3.8}$$

Using these identities, equations (5.3.4) and (5.3.5) are equivalent to

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{5.3.9}$$

$$\frac{\partial \rho c_1}{\partial t} = -\nabla \cdot (\rho \vec{u} \cdot c_1) + \rho \kappa_{c_1} \nabla^2 c_1 \tag{5.3.10}$$

Adding the equations for conservation of momentum and energy and using the equation of state $P = \frac{\rho R T}{\mu}$ of section 2.4.1, our two-component fluid is described completely by

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{5.3.11}$$

$$\frac{\partial (\rho c_2)}{\partial t} + \nabla \cdot (\rho \vec{u} \cdot c_2) - \rho \kappa_{c_2} \nabla^2 c_2 = 0 \tag{5.3.12}$$

$$\frac{\partial (\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u} + P + \sigma) = \rho g \tag{5.3.13}$$

$$\frac{\partial (\rho e_{tot})}{\partial t} + \nabla \cdot ((E_{tot} + P)\vec{u} - \vec{q} - \sigma \vec{u}) = \rho g \tag{5.3.14}$$

5.3.1 The Fractional Step Algorithm for the extended Navier-Stokes Equations

We want to apply the fractional step method of section 5.3 to solve the extended Navier-Stokes equations. The advective flux $F_1(U)$ is now given by

$$F_1(U) = \begin{pmatrix} \rho u \\ \rho c_2 u \\ \rho u^2 \\ \rho e_{tot} u \end{pmatrix}$$

The Jacobian DF_1 is

$$DF_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ c_2 u & u & c_2 & 0 \\ -u^2 & 0 & 2u & 0 \\ -\frac{\rho e_{tot} u}{\rho} & 0 & \frac{\rho e_{tot}}{\rho} & u \end{pmatrix}$$

The eigenvalues of $DF_1(U)$ are all identical u, so we apply the Complementary Projection Method without transforming onto the characteristic space.
Since $F_2(U)$ and F_3 have not changed, we use the algorithm derived for the Navier-Stokes Equations.

5.3.2 Stability Criterion for Semiconvection

This is taken from [7].
To model diffusive convection in stars, we need to take into account that the star consists of a mixture of hydrogen and helium. So there exist two different fields (T, μ),μ denoting the **mean molecular weight**, with very different kinematic viscosities. The Schwarzschild criterion of section 5.2.1 still applies, but additionally we have a μ-gradient to consider:

$$\nabla_\mu = \frac{\partial ln(\mu)}{\partial ln(P)}$$

A positive ∇_μ corresponds to a mean molecular weight that is large at the center and low at the surface.
We define a stability parameter R_μ as

$$R_\mu = \frac{\nabla_\mu}{\nabla - \nabla_{ad}}$$

Thus criteria for semiconvective instability are deduced:

Semiconvection	$\nabla - \nabla_{ad} > 0$, $\nabla_\mu > 0$	$R_\mu > 0$
Ledoux unstable	$\nabla_\mu < \nabla - \nabla_{ad}$	$R_\mu < 1$
Ledoux unstable	$\nabla_\mu > \nabla - \nabla_{ad}$	$R_\mu > 1$

A similar process plays an important role in oceanography; here we have a temperature and a salinity field, (T, S). When both T and S increase from the ocean surface towards

the bottom, the result is cold, fresh water over warm, salty water. The S field is stable, the T field is unstable (heavy at the top) and one has diffusive convection. This phenomenon is called **salt fingers**.

Salt fingers	$\nabla - \nabla_{ad} < 0,\ \nabla_\mu < 0$	$R_\mu > 0$
Ledoux unstable	$\lvert\nabla_\mu\rvert > \nabla - \nabla_{ad}$	$R_\mu < 1$
Ledoux unstable	$\lvert\nabla_\mu\rvert < \nabla - \nabla_{ad}$	$R_\mu > 1$

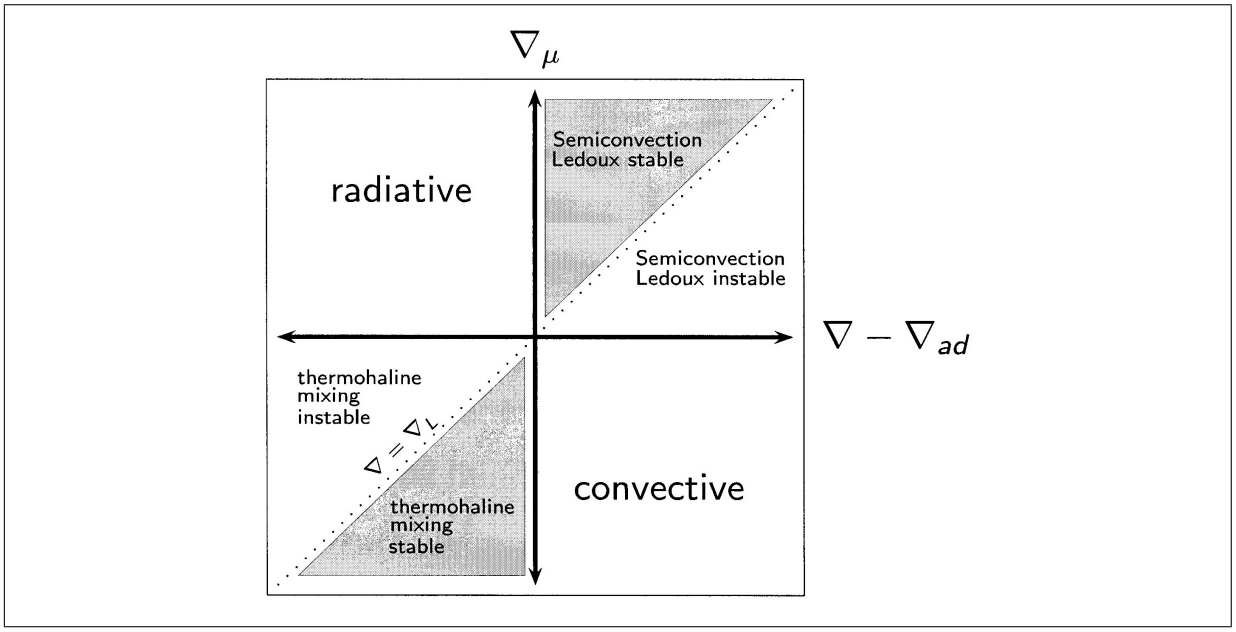


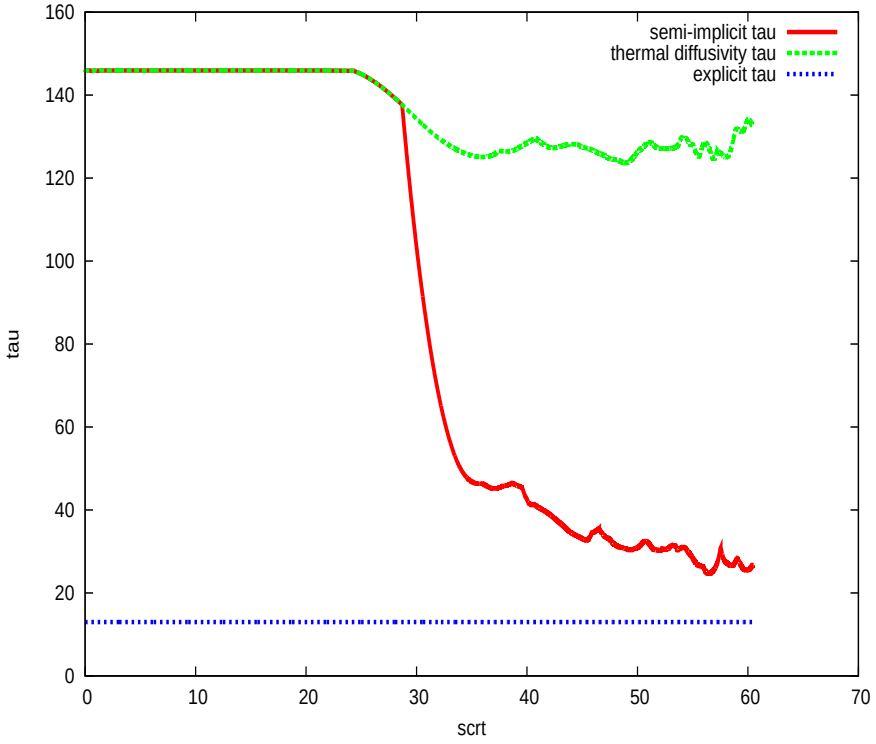
Figure 5.4: Stability criterion for semiconvection (Zaussinger,F. [1])

5.3.3 Simulations

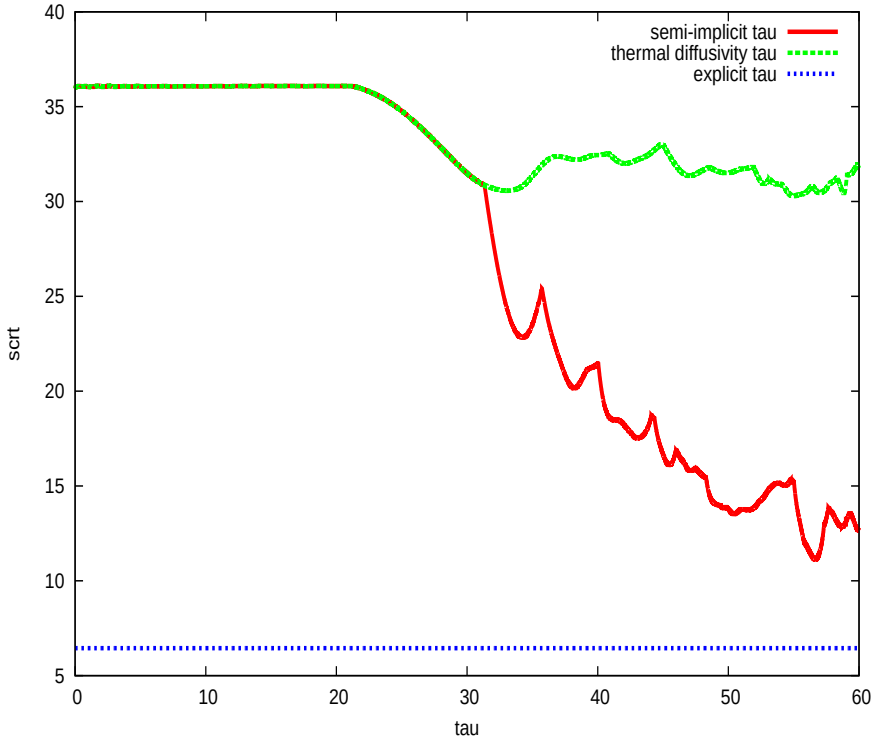
Apart from considering the mean molecular weight μ, the physical setting is analogous to that of the convective scenarios. The initial set-up was implemented by Zaussinger, F. [1].

We consider a fluid consisting of 50% hydrogen and 50% helium and play with the stability parameter R_μ. First, we simulate a convective scenario with $R_\mu = 0.2$.

In the second simulation we have $R_\mu = 1.1$.



(a) 100x101 grid points



(b) 200x201 grid points

Figure 5.5: The evolution of the timesteps in the simulations with $R_\mu = 1.1$. The behaviour is quite analogous to the one-component convective scenario.

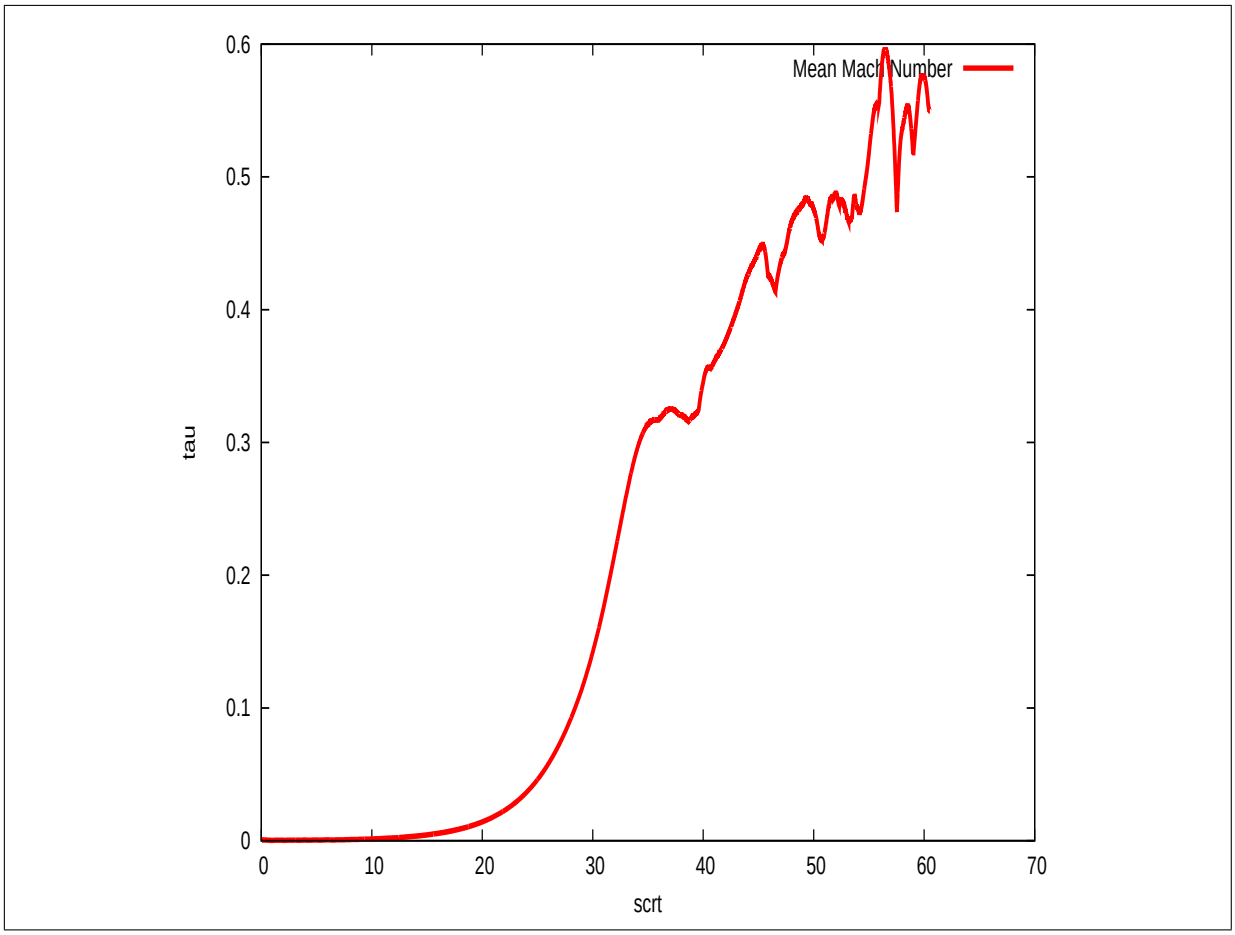
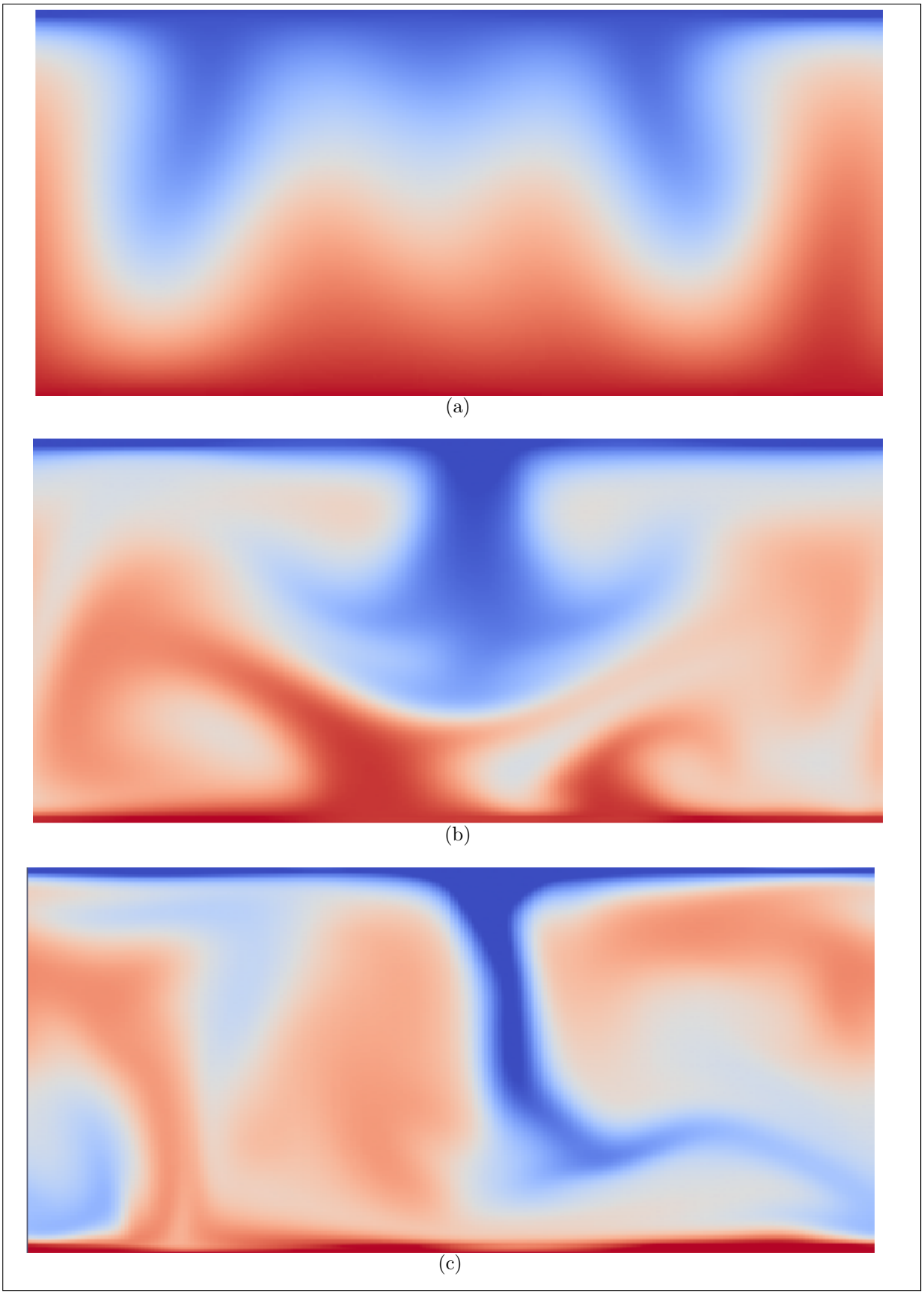


Figure 5.6: Mean Mach Number in the simulations with $R_\mu = 1.1$.

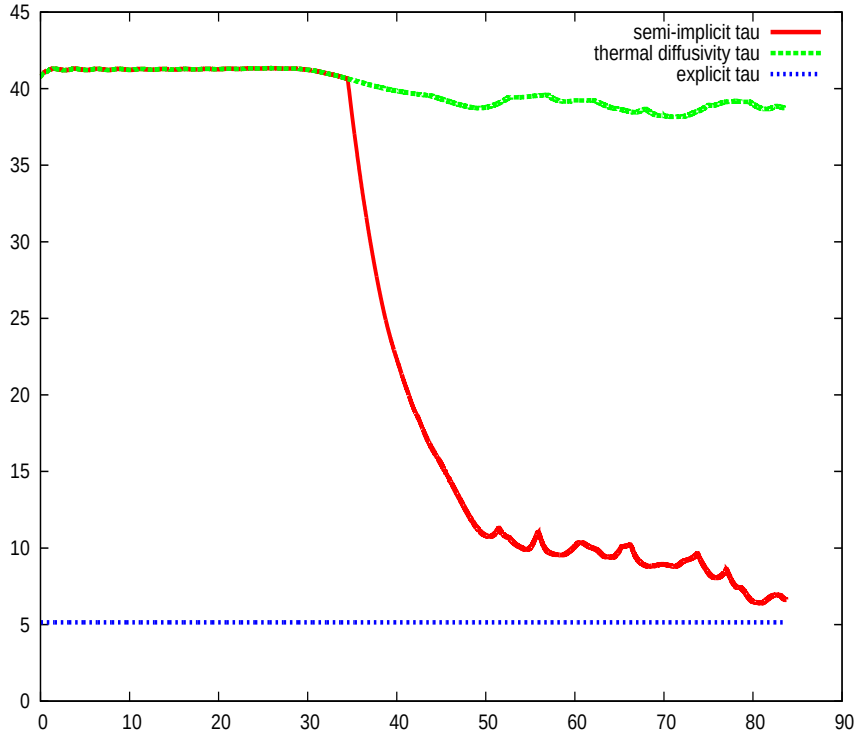


(a)

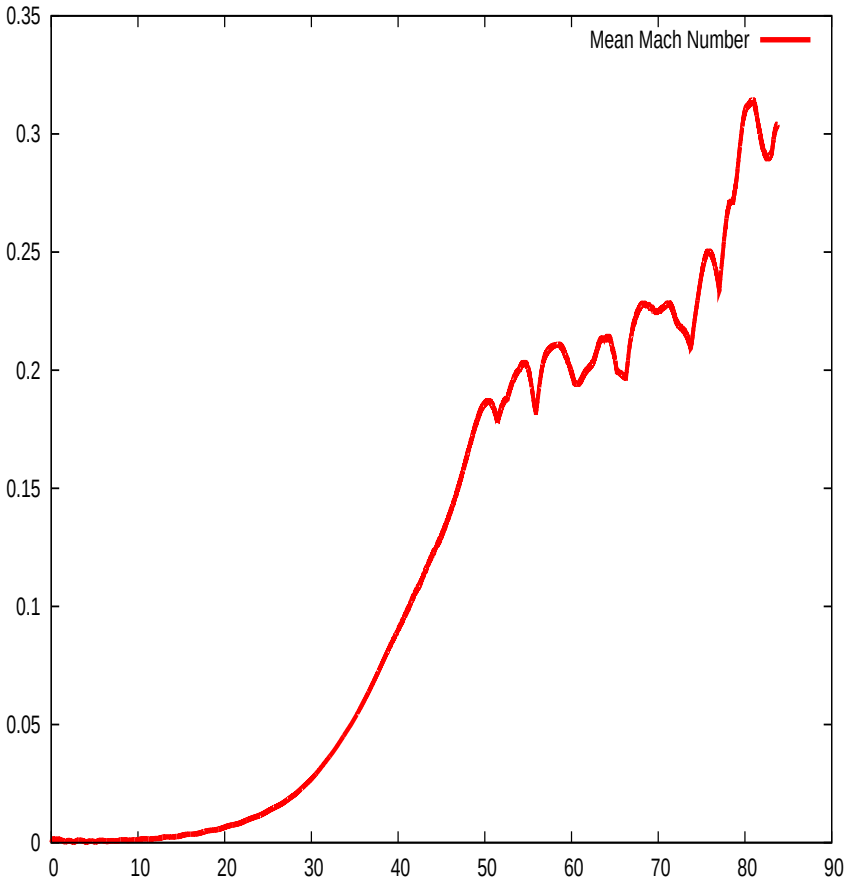
(b)

(c)

Figure 5.7: Snapshots of the simulation with 200x201 grid points. $R_\mu = 0.2$. Plotted is the potential temperature.

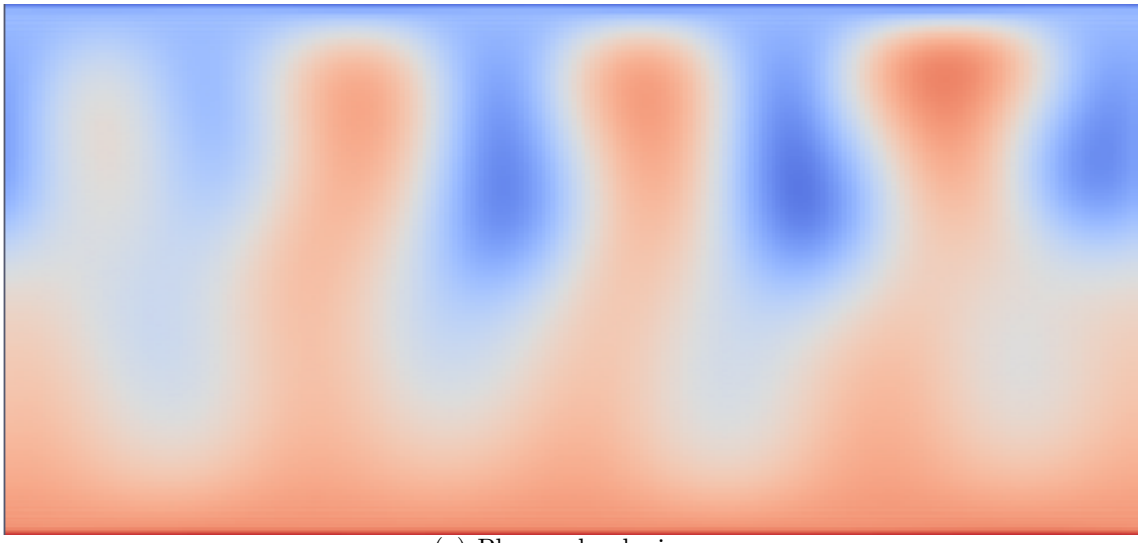


(a) The evolution of the timesteps in the simulations with $R_\mu = 0.2$.

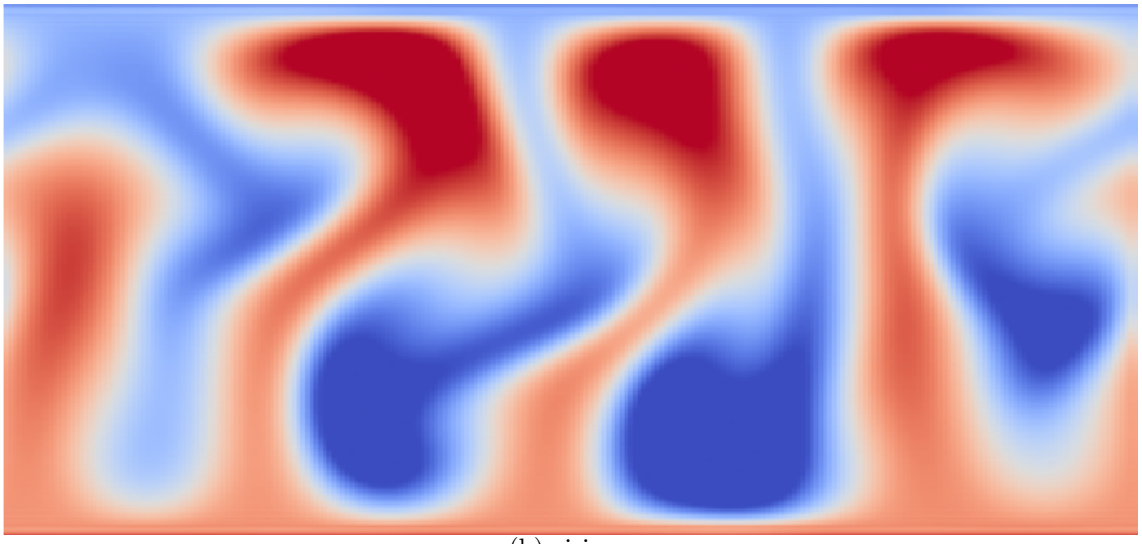


(b) Mean Mach Number in the simulations with $R_\mu = 0.2$.

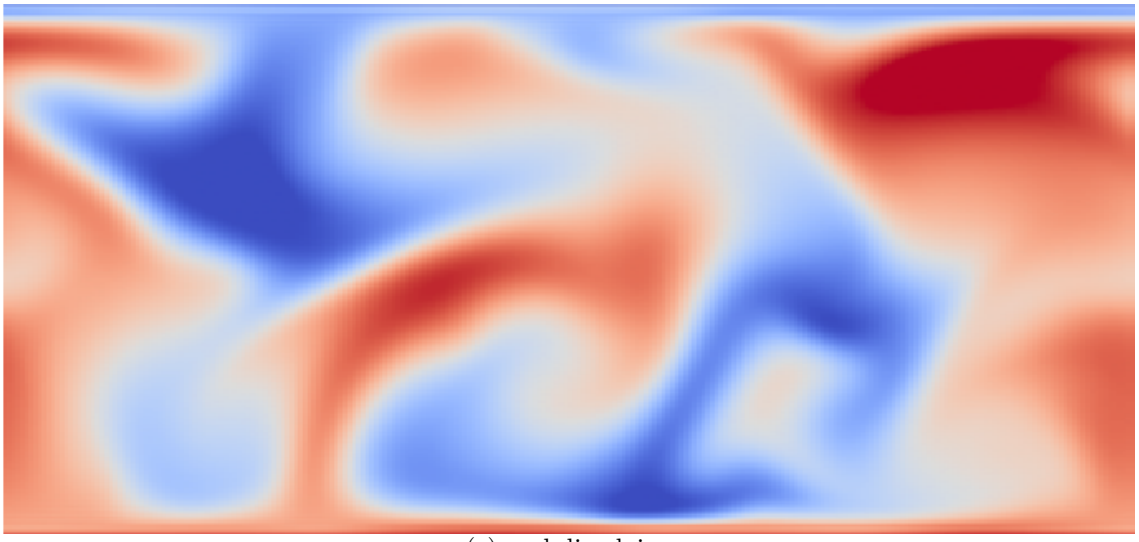
Figure 5.8:



(a) Plumes developing ..



(b) rising ..



(c) and dissolving.

Figure 5.9: Snapshots of the semiconvection simulation with 200x201 grid points. $R_\mu = 1.1$. Plotted is the potential temperature.

Appendix A

A.1 Notational Issues

Consider a scalar-valued function $g : \mathbb{R}^3 \to \mathbb{R}$, and a vector field $f : \mathbb{R}^3 \to \mathbb{R}^3$. Both be continously differentiable.

- ∇g denotes the gradient of g: $\nabla g = (\frac{\partial}{\partial x_1} g, \frac{\partial}{\partial x_2} g, \frac{\partial}{\partial x_3} g)^t$
- $\nabla \cdot f$ denotes the divergence of f: $\nabla \cdot f = \frac{\partial}{\partial x_1} f + \frac{\partial}{\partial x_2} f + \frac{\partial}{\partial x_3} f$
- ∇f denotes the Jacobian matrix of f: $\nabla f = (\frac{\partial f_i}{\partial x_j})_{ij}$

A.2 Gauss's Divergence Theorem

Theorem 1.2.1. *Be W a region in space with boundary ∂W and F a continuously differentiable vector field. Then the volume integral of the divergence $\nabla \cdot F$ of F over W and the surface integral of F over the boundary ∂W of W are related by*

$$\int_W \nabla \cdot F dV = \int_{\partial W} F \cdot \vec{n} dS \tag{A.2.1}$$

$\vec{n}$ is the outward pointing unit normal field of the boundary ∂W.

Proof. A proof is given in [12] □

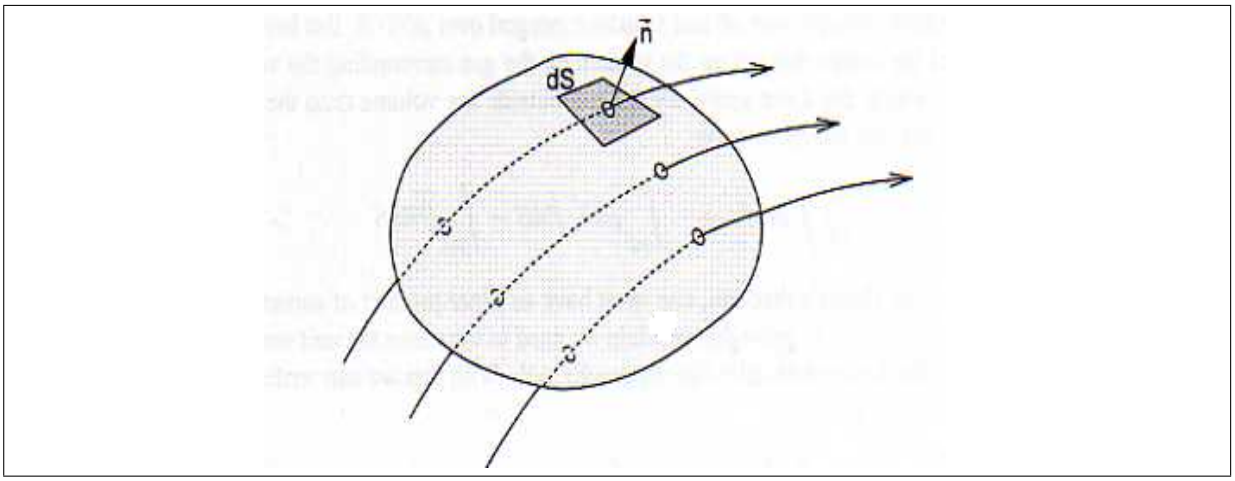


Figure A.1: Flow through a volume with surface S. The diagram shows Gauss's theorem by which the change of the total content within the volume equals the integral of the flux through its surface.

A.3 Comoving regions

Be $W(0)$ a subregion of D at time $t_0 = 0$. As time transcurres, the particles in W travel. We descruve their displacement by the function

$$\mathtt{x} \to \xi(\mathtt{x}, t)$$

Consequently, the region W at time t is defined as:

$$W(t) = \{\xi(\mathtt{x}, t) \mid \mathtt{x} \in W(0)\} =: \xi(W, t) \tag{A.3.1}$$

Be $V(t)$ the volume of $W(t)$. We obviously find that

$$V(t) = \textstyle\int_{W(t)} d\xi$$

Integration by substitution yields

$$V(t) = \int_{W(0)} J(x,t)dx \tag{A.3.2}$$

where $J(x,t) = det(\frac{\partial \xi_i}{\partial x_j})$ is the corresponding functional determinant.

Lemma 1.3.1. *The functional determinant $J(x,t)$ satisfies the following property:*

$$\frac{\partial}{\partial t} J(x,t) = J(x,t)(\nabla \cdot \vec{u}) \tag{A.3.3}$$

Proof. We follow the particle at the initial position $\mathtt{x}$ along its trajectory:

$$\frac{\partial}{\partial t}\xi(\mathtt{x},t) = \vec{u}(\xi(\mathtt{x},t),t) \tag{A.3.4}$$

Using the definition of the determinant gives

$$J(\mathtt{x},t) = det(\frac{\partial \xi_i}{\partial x_j}) = \tag{A.3.5}$$

$$\sum_{\sigma\in\mathfrak{S}_n} sign(\sigma)\partial_{\sigma(1)}\xi_1 \cdots \partial_{\sigma(n)}\xi_n \tag{A.3.6}$$

The sign of the permutation σ is defined as $sign(\sigma) = (-1)^r$, where r is the number of transpositions the permutation is composed of. $\mathfrak{S}_n$ denotes the symmetric group of degree n.

We now compute the temporal derivative of $J(\mathtt{x},t)$:

$$\frac{\partial}{\partial t}J(\mathtt{x},t) = \sum_{i=1}^{n} \underbrace{\sum_{\sigma\in\mathfrak{S}_n} sign(\sigma)\partial_{\sigma(1)}\xi_1 \cdots \partial_t\partial_{\sigma(i)}\xi_i \cdots \partial_{\sigma(n)}\xi_n}_{K_i(\mathtt{x},t)} \tag{A.3.7}$$

By definition of the determinant we see that

$$K_i(\mathtt{x},t) = det\begin{pmatrix} \partial_1\xi_1 & \cdots & \partial_t\partial_1\xi_i & \cdots & \partial_1\xi_n \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \partial_n\xi_1 & \cdots & \partial_t\partial_n\xi_i & \cdots & \partial_n\xi_n \end{pmatrix} \tag{A.3.8}$$

Using equation (A.3.4) and the symmetry of the second derivatives, we get

$$\partial_t\partial_j\xi_j = \partial_j\partial_t\xi_i = \partial_j u_i(\xi(\mathtt{x},t),t) = \sum_{k=0}^{n} \partial_k u_i \partial_j \xi_k \tag{A.3.9}$$

We insert this into (A.3.8)

$$K_i(\mathtt{x},t) = \sum_{k=0}^{n} \partial_k u_i det\begin{pmatrix} \partial_1\xi_1 & \cdots & \partial_1\xi_k & \cdots & \partial_1\xi_n \\ \vdots & & \vdots & & \vdots \\ \partial_n\xi_1 & \cdots & \partial_n\xi_k & \cdots & \partial_n\xi_n \end{pmatrix} \tag{A.3.10}$$

Contemplating the sum on the right hand side of equation (A.3.10), we see that if the indices k and i differ, there are two equal rows in the matrix $K_i(\mathtt{x},t)$. Consequently, the

determinant becomes zero, so equation (A.3.10) reduces to

$$\sum_{k=0}^{n} \partial_k u_i det \begin{pmatrix} \partial_1\xi_1 & \cdots & \partial_1\xi_k & \cdots & \partial_1\xi_n \\ \vdots & & \vdots & & \vdots \\ \partial_n\xi_1 & \cdots & \partial_n\xi_k & \cdots & \partial_n\xi_n \end{pmatrix} = \partial_i u_i J(\mathtt{x}, t) \tag{A.3.11}$$

Combining equations (A.3.7) and (A.3.11) we get

$$\partial_t J(\mathtt{x}, t) = \nabla \vec{u} \cdot J(\mathtt{x}, t) \tag{A.3.12}$$

which is exactly what we proposed. □

Theorem 1.3.2. Transport Theorem

Be W an arbitrary region in space, $\rho(\mathtt{x}, t)$ the mass density as defined in Chapter 2, and $f : W \times (0, \infty) \to \mathbb{R}$ continously differentiable. Then

$$\frac{d}{dt}\int_{W(t)} \rho(\xi, t) f(\xi, t) d\xi = \int_{W(t)} \rho \frac{Df}{Dt}(\xi, t) d\xi$$

holds.

Proof. First, we transform onto the initial region $W(0)$ using integration by substitution:

$$\frac{d}{dt}\int_{W(t)} \rho(\xi, t) f(\xi, t) d\xi =$$
$$\frac{d}{dt}\int_{W(0)} \rho(\xi(\mathtt{x}, t), t) f(\xi(\mathtt{x}, t), t) J(\mathtt{x}, t) dx$$

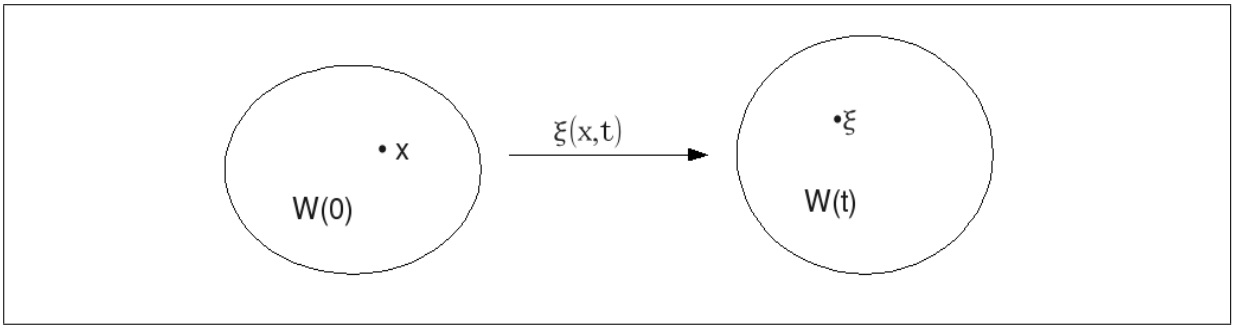


Figure A.2: Transformation of $W(0)$ to the comoving region $W(t)$

Since we assume ρ and f to be continuous, it is justified to move the differential operator

into the integral:

$$\frac{d}{dt}\int_{W(0)} \rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t) J(\mathtt{x},t) dx =$$
$$\int_{W(0)} \frac{d}{dt}(\rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t) J(\mathtt{x},t) dx =$$
$$\int_{W(0)} \frac{d}{dt}(\rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t)) J(\mathtt{x},t) + \rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t) \nabla\vec{u} J(\mathtt{x},t) dx =$$
$$\int_{W(0)} (\frac{d}{dt}(\rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t)) + \rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t) \nabla\vec{u}) J(\mathtt{x},t) dx =$$

Transforming back to $W(t)$ and using the material derivative defined in definition 2.2.1 leads to

$$\int_{W(0)} (\frac{d}{dt}(\rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t)) + \rho(\xi(\mathtt{x},t),t) f(\xi(\mathtt{x},t),t) \nabla\vec{u}) J(\mathtt{x},t) dx =$$
$$\int_{W(t)} \frac{D}{Dt}(\rho(\xi,t) f(\xi,t)) + \rho(\xi,t) f(\xi,t) \nabla\vec{u} d\xi$$

We remember the continuity equation (2.4.1):

$$0 = \frac{\partial\rho}{\partial t} + div(\rho\vec{u}) = \frac{\partial\rho}{\partial t} + \rho\nabla\cdot\vec{u} + \vec{u}\cdot\nabla\rho = \frac{D\rho}{Dt} + \rho\nabla\cdot\vec{u}$$

Thus

$$\int_{W(t)} \frac{D}{Dt}(\rho(\xi,t) f(\xi,t)) + \rho(\xi,t) f(\xi,t) \nabla\vec{u} d\xi = \int_{W(t)} \rho\frac{Df}{Dt} d\xi$$

□

Theorem 1.3.3. *Be $f : W \times (0,\infty) \to \mathbb{R}$ continously differentiable. Then*

$$\frac{d}{dt}\int_{W(t)} f(\xi,t) d\xi = \int_{W(t)} \left(\frac{\partial f(\xi,t)}{\partial t} + \nabla\cdot(f(\xi,t)\vec{u}(\xi,t)\right) d\xi \tag{A.3.13}$$

holds.

Proof. We use integration by parts and Lemma (1.3.1):

$$\frac{d}{dt}\int_{W(t)} f(\xi,t)d\xi = \frac{d}{dt}\int_{W(0)} f(\xi(\mathtt{x},t),t)J(\mathtt{x},t)d\mathtt{x} =$$
$$\int_{W(0)}\left(\frac{Df(\xi(\mathtt{x},t),t)}{Dt}J(\mathtt{x},t) + f(\xi(\mathtt{x},t),t)\frac{dJ(\mathtt{x},t)}{dt}\right)d\mathtt{x}) =$$
$$\int_{W(0)}\left(\frac{Df(\xi(\mathtt{x},t),t)}{Dt} + f(\xi(\mathtt{x},t),t)\nabla\cdot\vec{u}\right)J(\mathtt{x},t)d\mathtt{x} =$$
$$\int_{W(t)}\left(\frac{\partial f(\xi,t)}{\partial t} + \nabla\cdot(f(\xi,t)\vec{u})\right)d\xi$$

□

Bibliography

[1] 2009-2010. F. Zaussinger. Private Communication. (PhD-student, Max-Planck Institute for Astrophysics, Munich, Germany).

[2] A.Weiss, W.Hillebrandt, H.-C.Thomas, and H.Ritter. *Cox & Giuli's Principles of Stellar Structure*. Cambridge Scientific Publishers Ltd, extended second edition, 2004.

[3] Lawrence C. Evans. *Partial Differential Equations*. American Mathematical Society, 2002.

[4] Sigal Gottlieb and Chi-Wang Shu. Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, 67(221):73–85, January 1998.

[5] Hannes Grimm-Strele. Numerical solution of the generalised poisson equation on parallel computers. Master's thesis, Universität Wien, März 2010.

[6] J.-L. Guermond and Abner Salgado. A splitting method for incompressible flows with variable density based on a pressure poisson equation. *Journal of Computational Physics*, 228:2834–2846, 2009.

[7] Wolfgang Hillebrandt and Friedrich Kupka (Editors). *Lecture Notes in Physics 756: Interdisciplinary Aspects of Turbulence*. Springer Verlag Berlin Heidelberg, 2009.

[8] H.J.Muthsam, F.Kupka, B. Löw-Baselli, C.Obertscheider, M.Langer, and P.Lenz. Antares - a numerical tool for astrophysical research with applications to solar granulation. *New Astronomy*, 15:7460–475, 2010.

[9] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[10] Nipun Kwatra, Jonathan Su, Jón T. Grétarsson, and Ronald Fedkiw. A method for avoiding the acoustic time step restriction in compressible flow. *Journal of Computational Physics*, 228:4146–4161, June 2009.

[11] L.D. Landau and E.M. Lifschitz. *Lehrbuch der theoretischen Physik 4, Hydrodynamik*. Akademie-Verlag Berlin, 4. edition, 1981.

[12] Jerrold E. Marsden and Anthony Tromba. *Vector Calculus*. Palgrave Macmillan, 5 edition, 2003.

[13] N. Mauser and C. Heitzinger. Lecture notes on applied analysis, Summer term 2009.

[14] H.J. Muthsam. Lecture notes on hydrodynamics, Summer term 2009.

[15] H.J. Muthsam, W. Göb, F. Kupka, and W. Liebich. Interacting convection zones. *New Astronomy*, 4:405–417, 1999.

[16] A. Neumaier. *Introduction to Numerical Analysis*. Cambridge, 2001.

[17] Christoph Obertscheider. Essentially Non-Oscillatory Verfahren zur numerischen Lösung der Diffusionsgleichung. Master's thesis, Universität Wien, November 2005.

[18] Barry Merriman Ronald P. Fedkiw and Stanley Osher. Efficient characteristic projection in upwind difference schemes for hyperbolic systems. *Journal of Computational Physics*, 141:22–36, 1998.

[19] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. *ICASE Report No. 97-65*, 1997.

[20] John A. Trangenstein. *Numerical Solution of Hyperbolic Partial Differential Equations*. Cambridge University Press, 2009.

Curriculum Vitae

Name:	Natalie Happenhofer
Birth:	22nd of October, 1986, in Vienna, Austria
Citizenship:	Austria
Driving Licence:	B
Foreign languages:	English, Spanish
Programming languages:	Fortran90, C++, MatLAB

Education

September 1993 - June 1997:	Primary school: Volksschule Weikersdorf in Baden bei Wien
September 1997 - June 2005:	High school: Bundesgymnasium Baden, Frauengasse 3-5 in Baden bei Wien, Austria
June 2003 :	Cambridge First Certificate of English
August 2003 - July 2004 :	Student exchange to Lima, Peru
June 2005:	High school diploma with distinction
October 2005 - June 2010	Diploma studies in Mathematics

Working experience

September 2004 - March 2008:	Employee at the supermarket BILLA (12h/week)
October 2008 - Februray 2008:	Tutor for 'Analysis for Physicists' Institute for Mathematics, University of Vienna
March 2009 :	Carer at the exposition 'Imaginary' in Vienna organized by the Mathematical Institute of Oberwolfach, Germany
October 2009 - Februray 2010:	Tutor for 'Introduction to linear algebra' Institute for Mathematics, University of Vienna
Since February 2008:	Working student at the research group ACORE Institute for Mathematics, University of Vienna Field of Activity: Visualization of scientific data, especially extending the open source software ParaView

Zusammenfassung

In der vorliegenden Arbeit werden die Navier-Stokes Gleichungen für kompressible Fluide aus den physikalischen Erhaltungsgesetzen von Masse, Impuls und Energie hergeleitet. Dank dieses allgemeinen Ansatzes können die Navier-Stokes Gleichungen zur Modellierung unterschiedlichster physikalischer Problemstellungen herangezogen werden, angefangen von der Strömung durch ein Rohr in Ingenieurswissenschaften bis zu stellarer Konvektion in Astrophysik.

Mathematisch betrachtet bilden die Navier-Stokes Gleichungen ein beinahe hyperbolisches System partieller Differentialgleichungen, da wir Fluide mit geringer Zhigkeit betrachten. Verschiedene numerische Verfahren wurden entwickelt, um deren Lösung zu approximieren, speziell in dieser Diplomarbeit wird das Verfahren von N. Kwatra et al. (siehe [10]) verwendet. Diese Methode benutzt ein ENO-Verfahren (siehe [9], [18],[19]) zur Berechnung des advektiven Teils der Gleichungen und korrigiert die Zwischenwerte dann mittels der Lösung einer Poisson-ähnlichen Druckevolutionsgleichung.
Die Trennung von Advektions- und Drucktermen bewirkt, dass Schallwellen gefiltert werden. Schallwellen transportieren wenig Energie, breiten sich jedoch sehr schnell aus, sodass ihre Auflösung eine starke Einschränkung an die Zeitschrittweite mit sich bringt. Deren Filterung bewirkt, dass der Zeitschritt in N.Kwatras Methode signifikant größer gewählt werden kann als bei herkömmlichen Verfahren.

Insbesondere bei der Simulation von Fluiden, deren Geschwindigkeit nur einen Bruchteil der Schallgeschwindigkeit beträgt, ist der Zeitgewinn enorm. Dies wird an Simulationen von stellarer Konvektion und doppelt-diffusiver Konvektion in Sternen illustriert.